AvnFPS Documentation

June 19, 2003

[Return to AvnFPS1.0 Build Page]

Content

Introduction

The Aviation Forecast Preparation System (**AvnFPS**) is modeled after the **Morristown Aviation Workstation** written by David Hotz and Paul Kirkwood.

AvnFPS is designed to ease monitoring, improve production, and facilitate Quality Control (QC) of TAFs and TWEBs. For example, the AvnFPS monitoring capability gives forecasters quick and continuous feedback on TAFs as well as associated observations. This monitoring capability consists of a color coded scheme and is site configurable. The TAFs, the guidance, and the current observations are displayable as both text and graphics. The display of either text or graphics is controlled by the forecaster. The local office also has another feature when it comes to amendment criteria, the local office can configure each airport to display additional criteria such as cross winds and different cig/vis. This is known as Airport Specific TAF Amendment Criteria (ASTEC).

Eventually, AvnFPS will involve verification. A local file will be sent long line, so TAFs, observations, and guidance data all reside at NWS Headquarters. This will be part of the new stats on demand and is intended to ease the workload on the aviation focal points who now have the responsibility of entering data manually for individual forecasters.

Last Updated: Carl McCalla, Sr. *06/19/2003*
AvnFPS Software Developer: George Trojan (SAIC/MDL)
Technical Contributor: David Hotz (WFO MRX)

AVNFPS Overview

Release 1.0
June 19, 2003

Introduction

This document describes overall design of the system.

Most of **avnfps** is written in *Python*. *Python* is an interpreted language, which imposes a performance penalty on the operating system. The time critical parts, such as decoders are written in *C++*. The current release requires *Python 2.2* which is distributed with AWIPS build 5.2.2. The C++ extensions are built in the python executable, named avnpython, which is part of **avnfps** distribution.

Directory structure

The official release of **avnfps** will be installed in the directory /awips/adapt/avnfps. It is possible to change the installation directory, by changing the value of TOP_DIR in the shell wrapper scripts.

```
/awips/adapt/avnfps/
      bin/          see section Wrapper Scripts
            HP-UX/
            Linux/
      data/         see section Data Feed
      doc/
            html/
                  gifs/
      etc/          see section Configuration
      logs/
      xmit/         see section Transmission
      tmp/
      py/
      src/
```

For details on file formats refer to section AVNFPS File Format.

User Interfaces

There are two main graphical user interfaces (GUI) to **avnfps**:

- avnsetup.py: program to configure the system (bulletins issued by the office, appearance and behavior of the monitoring and forecast preparation units.
- avnwatch.py: this is a TAF monitoring program. The forecast preparation unit is implemented in a form of dialogs that are started from the main GUI.
  avnwatch.py and avnsetup.py are normally run on a workstation (HP or Linux).
- avnmenu.py: allows to select forecaster id before starting avnwatch. That way the (optional) X resources unique for the particular forecaster are known and can be processed on the startup of the monitoring GUI.
- avnxmitserv.py: a transmission server that runs on the data server (currently HP only). The server is started from avnwatch.py.

All GUIs are implemented in *Tkinter* and *Pmw*. *Tkinter* is a built−in Python implementation of *Tk*. That is, all the windows are instantiated by *Tk* code and consequently their behavior is governed by *Tk* design. *Pmw* (Python mega widgets) is a set of widgets implemented in Tkinter.

avnsetup.py, avnmenu.py and avnsetup.py are stand−alone entities. All other windows are implemented as dialogs. All dialogs except error/warning/info messages are cached, created the first time when needed. When closed, the dialogs are unmapped, but not destroyed. This means there is no need to be able to iconify a dialog. Once a dialog is created, it is as efficient to press the button that invokes the dialog, as would be de−iconify the window.
The main window and the dialogs form a hierachy. With the usual window manager policy, the dialog that is instantiated from the main window or another dialog always stays on the top of it. Independent dialogs can have their stacking order changed. Depending on the window manager settings, this can be done either by pointing or clicking on the respective window. It is possible to change the window manager policy. The procedure will vary depending on the window manager.
On HP worstations, the value to change is *Allow Primary Windows On Top*, available from *Window* submenu of *Style Manager/Window*.
On Linux machines, use *GNOME configuration tool*, then select *Miscellaneous* in *Sawfish window manager*. The value to change is *Keep transient windows stacked above*.
Keep in mind that:

- Changing the settings affects all the applications. This may be what you want though, to preserve consistency.
- The drawback is that some windows requiring response, called *modal dialogs* may become hidden under other windows. The whole application will be seen as unresponsive to any user actions ("hung").

An alternative to changing window manager's behavior is to set the value of *transientDialogs: option in the X resources file.

Wrapper Scripts

Since HP and Linux architectures require distinct Python binaries, the programs above are wrapped in Bourne shell scripts that set the binaries properly. There are 4 scripts:

- avnsetup.sh starts avnsetup.py GUI
- avnwatch.sh starts avnmenu.py GUI. avnwatch.py is called from the menu.
- avnxmitserv.sh starts avnxmitserv.py. This script is normally started from the TAF monitoring GUI.
- runtrig.sh a utility program to populate **avnfps** database during system setup.

There are four environmental variables defined in the wrapper scripts:

- `export TOP_DIR=/awips/adapt/avnfps`
  TOP_DIR points to the top level directory where **avnfps** is installed. Do not modify this one, as the trigger programs have the path hardcoded.
- `export FXA_HOME=/awips/fxa`
  This will never change.
- `export PYTHONHOME=/usr/local/python`
  Theoretically, **avnfps** may require a different release of Python than the rest of AWIPS. The current value is correct for AWIPS 5.2.2+ and this version of **avnfps**.
- `export BROWSER=netscape`
  This variable is optional. It tells the main GUIs which web browser to call to display this help. If BROWSER is not defined on Linux systems, mozilla will be chosen.

Data Feed

avnfps keeps it own database for METAR, TAF and TWB data. The database consists of ASCII data files located in a directory tree under /awips/adapt/avnfps/data/:

```
/awips/adapt/avnfps/
     data/
           mtrs/
                 KDCA/
                      0110110520=
                       . . . .
                 . . . .
           tafs/
                 KDCA/
                      0110111124=
                       . . . .
                  . . . .
           twbs/
                 073/
                        0110111414=
                         . . . .
                  . . . .
```

The data are received via the AWIPS trigger mechanism. This works as follows:
The list of requested products and corresponding programs is stored in Informix database **fxatext**, table **watchwarn**. For each product that is listed in the table, AWIPS ingest software writes it also as an ASCII file to the directory /data/fxa/trigger/. An associated Informix procedure executes the corresponding program passing the bulletin's file name as an argument.
The setup program, avnsetup.py is used to create the **avnfps** trigger file, /awips/adapt/avnfps/etc/triggerTemplate.

The AWIPS localization program, /awips/fxa/data/localization/scripts/mainScript.csh merges the template file with other AWIPS application trigger definitions and updates watchwarn table.

There are three data types in **avnfps** database, for METARs, TAFs and TWBs (AVN MOS data is handled differently, see below). Each requires a separate program: these are, respectively, mtrtrigger, taftrigger and twbtrigger located in /awips/adapt/avnfps/bin/HP−UX/. These programs extract header time, report type and station id (or route for TWBs) and write unmodified data to files as shown in the above diagram. The file name is in the format `yymmddHHMMx` where x is:

- = for regular or delayed forecast
- A for amendment
- C for correction.

The trigger programs are responsible for cleanup of old bulletins. When a new bulletin is written to the respective directory, all bulletins older than 25 hours are deleted.

The success or failure of triggers and decoders are logged. See the Logging section for the details.

Aviation MOS data is retrieved from AWIPS NetCDF files. These are stored in /data/fxa/point/mos/AVN/netcdf/. Text bulletins, similar to MAVs are generated by the display program.

Transmission

```
/awips/adapt/avnfps/
      xmit/
            pending/
            sent/
            bad/
```

The forecasts are transmitted by the program avnxmitserv.py which runs on the data server. This is a Python script. For technical reasons, it is started via a shell script, avnxmitserv.sh, located in /awips/adapt/avnfps/bin/. The forecasts prepared for transmission are written to directory /awips/adapt/avnfps/pending/. The name of the transmission file is in the following format:

```
001-KLWXTWB032-FRUS41-KLWX-0302041900-AAA-1044376472
```

where

- `001` is the forecaster number
- `KLWXTWB032` is the AWIPS id
- `FRUS41-KLWX-0302041900` is the WMO header
- `AAA` indicates an amendment. For regular issue forecasts, this field would be ___ (3 underscores).
- `1044376472` is the earliest transmission time (seconds from Jan 01, 1970)

avnxmitserv.py scans the directory /awips/adapt/avnfps/xmit/pending/ every 30 seconds. This can be changed, see Configuring avnxmitserv below.

Each file name is parsed; if it does not match the pattern above, the file is moved to the directory /awips/adapt/avnfps/xmit/bad/. A similar fate awaits file older than 3 hours. If the file contains a regularly issue forecast (i.e. the suffix is ___), the program checks whether current time falls within the transmission window. If not, the file is skipped.

If the file passess all the checks, avnxmitserv.py executes the AWIPS transmission program, handleOUP.pl to transmit the forecast.
The return code is examined and transmission status is logged. See the Logging section for details.

After successful transmission, the forecast file is moved to the directory /awips/adapt/avnfps/xmit/sent/. When it starts or at the first run after 00Z, avnxmitserv.py deletes files from /awips/adapt/avnfps/xmit/bad/ and /awips/adapt/avnfps/xmit/sent/ that are older than 2 days.

Configuring avnxmitserv

To modify behavior of avnxmitserv.py, edit the wrapper script avnxmitserv.sh. The line to modify is:

```
exec ./avnpython $TOP_DIR/py/avnxmitserv.py
```

As delivered, the program is configured for individual forecast transmission, with the frequency of scanning the pending directory set to 30 seconds. To change the frequency, use the argument −t timeout, where timeout is the frequency in seconds. An example: to set frequency to 20 seconds modify the above line to:

```
exec ./avnpython $TOP_DIR/py/avnxmitserv.py −t 20
```

It is possible to force a file to be transmitted. In that case, the forecast file name will have the transmit time (last field in the file name) updated and moved to /awips/adapt/avnfps/pending/.

avnxmitserv.py is usually started from the monitoring program, avnwatch.py, run on a workstation. The startup procedure checks whether another avnxmitserv.py is already running, in order to prevent multiple occurences. avnwatch.py also checks for the status of avnxmitserv.py. This is implemented trough a timestamp file /awips/adapt/avnfps/xmit/tstamp. This file is updated every time avnxmitserv.py reads /awips/adapt/avnfps/pending/.

avnxmitserv.py will quit after 10 unsuccessful tries. This indicates a system problem: network breakdown or, in setup stage, configuration problems, program bugs etc.

Note: avnxmitserv.py will not restart on data server failover. When the ds1 − ds2 switch occurs, you should start the server by pressing the **Last Pending Check** button.

Logging

avnfps uses circular method to store log files. All files are written to the directory /awips/adapt/avnfps/logs. The file name is: progname_Day where progname is the program name, and Day is 3−letter abbreviation of the day of the week. The programs that write log files are responsible for deletion of files older than 6 days.

Trigger Logs

Unless there is an error, entries in the log file are of the form

```
taftrigger 23:39:13: processed /awips/adapt/avnfps/data/tafs/KTPA/0211282330=
```

If a new site has been added to the trigger list, you may see a message such as

```
opendir(/awips/adapt/avnfps/data/twbs/002): No such file or directory
```

This is harmless: the directory will be created when needed. You may have a problem though when this message is repeated − the program fails for some reason (i.e. incorrect permissions) to create the new directory.

Transmission Logs

avnxmitserv.py writes to a file XmitServ_Day where Day is the UTC day of the week. Example content:

```
Starting at 15:19:10
FAIL  1: 23:31:58 001−KLWXTAFIAD−FTUS41−KLWX−0302032300−___−1044315000
SUCCESS: 23:32:31 001−KLWXTAFIAD−FTUS41−KLWX−0302032300−___−1044315000
```

The FAIL and SUCCESS are keywords. FAIL is succeeded by handleOUP.pl return code. This is always 1, or 127 if handleOUP.pl could not be executed for some reason.

SUCCESS lines are processed by the TAF and TWB editors in order to determine version numbers for non−regular issue forecasts.

Note: SUCCESS does not mean that the forecast has been delivered to the intended destination. It simply means that handleOUP.pl executed the AWIPS message handler. When you suspect that the forecasts are not transmitted, consult relevant logs in the fxa log directory /data/logs/fxa/YYYYMMDD/ on the data server.

Configuration

All the configuration files relevant to **avnfps** are located in the directory /awips/adapt/avnfps/etc/:

```
/awips/adapt/avnfps/
       etc/
               tafs/
               twbs/
               app−resources/
               bitmaps/
               tafelements/
               sounds/
```

The setup program also uses AWIPS configuration file /awips/fxa/data/afos2awips.txt to find AFOS headers required by the triggers.

To configure **avnfps** use the program avnsetup.sh. Most of the bulletin definition files, located in /awips/adapt/avnfps/etc/tafs/ and /awips/adapt/avnfps/etc/twbs/, are in binary format.

/awips/adapt/avnfps/etc/app−resources/ contains X resource configuration files (that is those that determine colors, fonts, windows sizes and program behavior). There should always be a file X containing common resources for the WFO. Individual forecasters can override these resources by creating their own files X.n, where **n** is forecaster id, as assigned in the setup program.

/awips/adapt/avnfps/etc/bitmaps/ contains weather element bitmap files used by the weather plot GUI.

/awips/adapt/avnfps/etc/tafelements/ contains lists of valid TAF element (WX, Wind etc...) values. The data in these files were extracted from the AWIPS hmdb database.

/awips/adapt/avnfps/etc/sounds/ is a placeholder for sound files. You may however use any sound file available on the workstation, as the absolute path is used in the X resources file.

Other files in /awips/adapt/avnfps/etc/:

- catalog.txt − catalog of error messages displayed by TAF QC procedure. This file should not be modified
- forecasters − site specific forecaster list file.

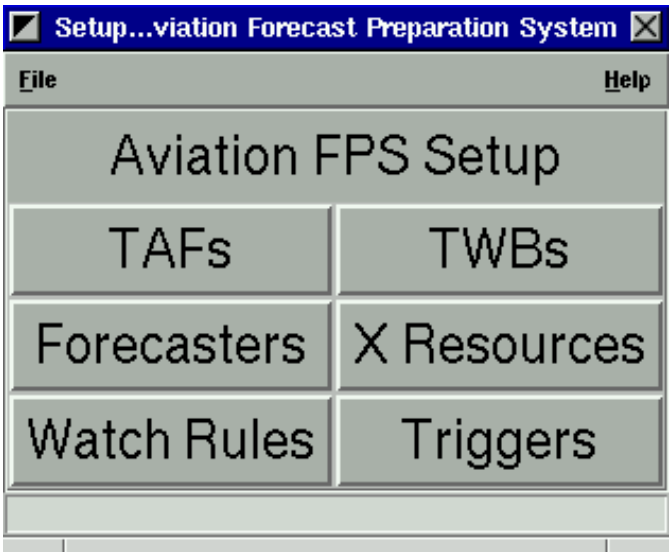
- triggerTemplate − this file is created by the setup program. It is processed by AWIPS localization script mainScript.csh.
- trigger.conf − file defining number of hours of reports stored in the data directory. This file is optional.
- climcheck.conf − contains various parameters used by the climate verification module.
- climcheck.conf − contains list of sites for which climate data are available.
- Various .jpg, .xbm and .gif files − used as icons by the window manager.

AVNFPS Setup

Release 1.0
June 19, 2003

How To Start avnsetup

The program is normally started from window manager menu. The details depend on the window manager: pop−up menu in dtwm (HP−UX), panel menu or a short−cut icon in Sawfish (Linux). In any case, the avnsetup.sh script is executed.

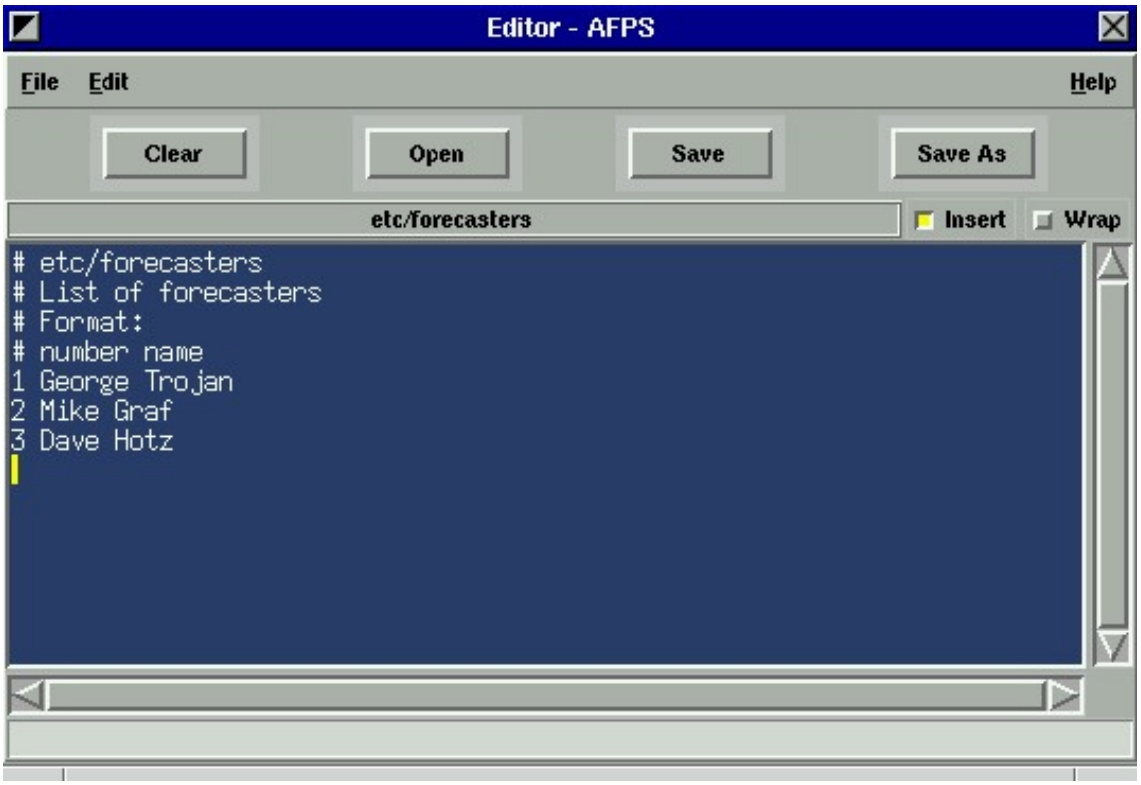


The main GUI is a menu for six separate entities (dialogs). These are described below. Each of them has available on−line help. Messages relevant to the operation are displayed according to their severity, either as a pop−up dialog, requiring confirmation, or in a status message bar just above the **Close** button in the dialogs, or the bottom on the main GUI.

In order to configure the software for your site, you need to complete the first four steps. The last two are optional. Click on **Forecasters**.

How To Create Forecasters List File

The forecaster list file, /awips/adapt/avnfps/etc/forecasters consists list of pairs (*number*, *forecaster name*). The number is used to form name for the file containing forecast to be transmitted. Transmitted forecasts are logged and the log files can be used for verification purposes. Select **Forecasters** button. A text editor with loaded forecaster list file will appear.



If the file does not exist, an empty one will be created:

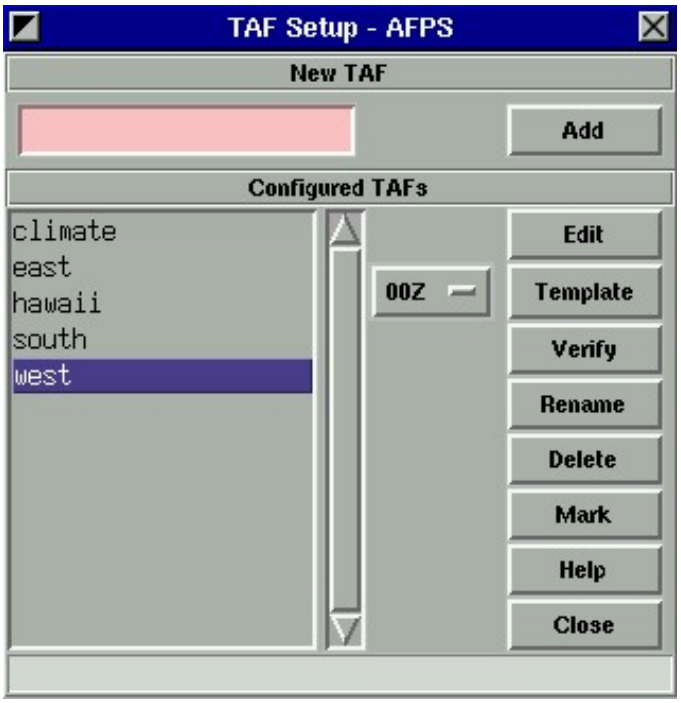
```
# etc/forecasters
# List of forecasters
# Format:
# number name
```

When complete, press the **Save** then the **Close** buttons. Use the **Help** option from the main menu to access on−line help.

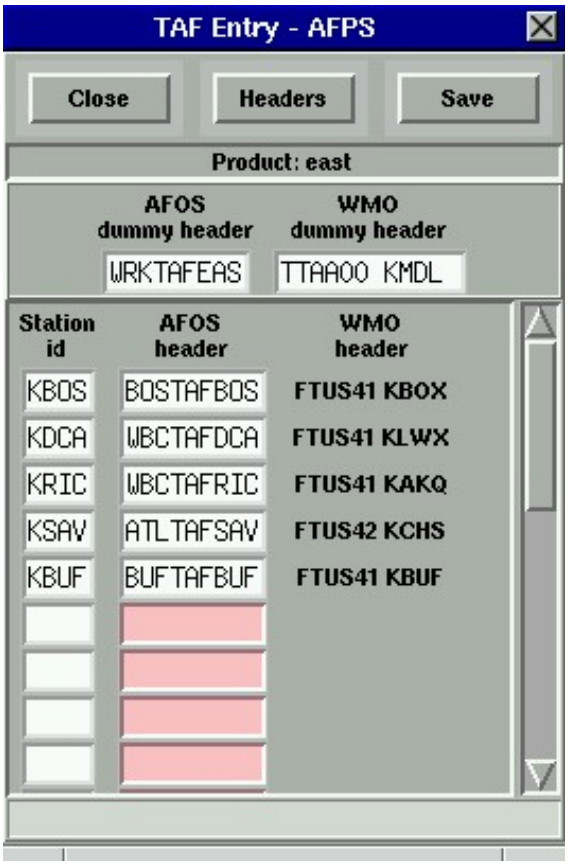
How To Configure TAF Bulletins

In normal circumstances, you should have one bulletin for all TAFs issued by your office and one for each office you may be asked to back up.

Select **TAFs** button. A TAF setup dialog will be displayed



The dialog contains two sections. The top one, consisting of an entry field and **Add** button is used to add new bulletin definitions. The bottom section, labelled **Configured TAFs** is used to view or modify existing bulletins. Enter a name for your main bulletin in the entry field. This can be any sequence of letters and digits separated by blanks.
Press the **Add** button. Another dialog will be shown.



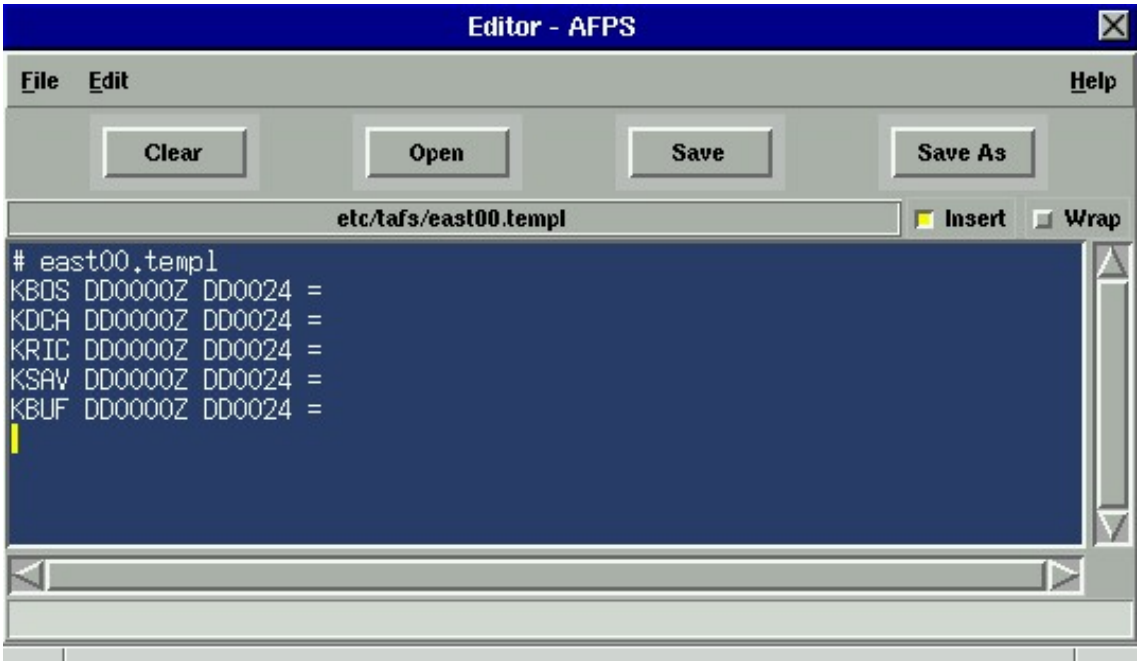
Enter site ids for all sites you want to include in the bulletin. Leave the **AFOS header** column blank.
Press the **Headers** button. The program will scan /awips/fxa/data/afos2awips.txt file for TAFs matching site ids entered. The corresponding headers will be displayed. The TAFs will be transmitted with WMO headers as shown.

Though the AFOS header entries are editable, unless you are completely certain of what you are doing, you should not modify the values. The AWIPS transmission program, handleOUP.pl, will refuse to transmit forecasts if the AFOS header cannot be found in afos2awips.txt.
Note: Since the directory /awips/fxa is local, it is possible that the afos2awips.txt file on a workstation may differ from the one on the server. In such case you may run into trouble.

The dummy headers are used to store work tafs in the text database. Those can be anything you want, provided that the standard format is followed, i.e. AFOS entry is 8 to 9 characters long.

Press the **Save** button. This action will first check whether the AFOS headers are valid. If not, an error dialog will show up and the procedure will abort. You will need to fix the headers (or the file afos2awips.txt).
If the headers are valid, several configuration files will be written to directory /awips/adapt/avnfps/etc/tafs. These are *yourlabel.def* and *yourlabel.xxx.list* where *yourlabel* is bulletin name chosen (blanks replaced by underscores) and *xxx* is *taf*, *mtr* and *mav*.
Also, skeleton template files, one for each regular issue forecast will be created: these are *yourlabelHH.templ*, where *HH* is *00*, *06*, *12* and *18*.
If there are no errors, you will see a *Save succeeded* message in the status window (just above the **Close** button). Press **Close** to close the dialog.
Note: You must enter both dummy headers for **Save** to succeed.

The bulletin label will show up in the **Configured TAFs** list. If you wish to modify templates, highlight the bulletin label, select issue hour from options menu and press the **Template** button. A text editor with the template file loaded will be displayed.



Make the changes and press **Save**, then **Close**. Repeat this step for each issue hour.

The entries are separated by an '=' sign or by a blank line. You may have an entry across several lines. Most likely you will want to modify entries for sites that do not report full 24 hours, for example:

```
KRWF DD2320Z DD0024 AMD NOT SKED 03Z-12Z=
```

The forecaster may then select the **Merge** load option in the TAF Editor GUI to initialize the forecast with the previous forecast, with the *AMD* ... phrase appended.

Finally, for each site defined in the bulletin, the program will copy default monitoring criteria file to the file CCCC.rules, (CCCC is the site id), also residing in /awips/adapt/avnfps/etc/tafs. These files can be customized via Monitoring Rules editor, described below.

The **Verify** button can be used to check for existence of all required bulletin definition files.

You may modify bulletin name. Highlight the old name, then press **Rename** button. This will pop up a dialog asking for the new name. Enter a name and press **OK**.

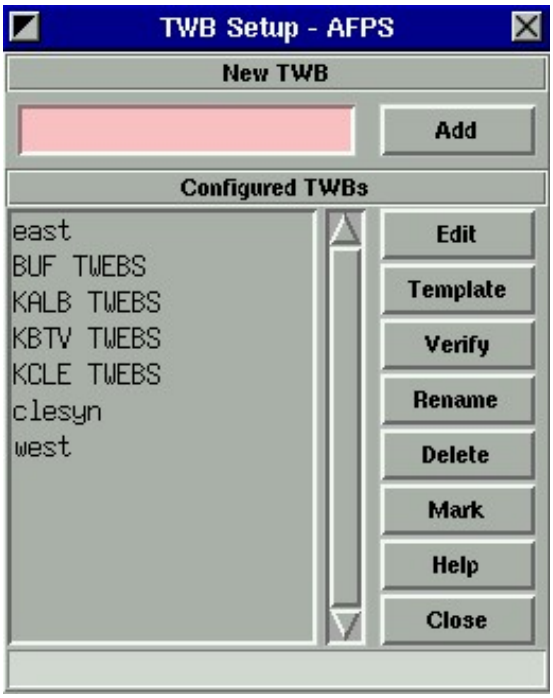
To delete an unwanted bulletin, highlight it in the **Configured TAFs** list window, then press **Delete**. This will (requires confirmation) delete all files described above, except the *.rules.

The bulletins are listed in alphabetical order. The sites listed in the first one are loaded on startup of avnwatch.py and displayed in the **Load** dialog of the forecast editor. If the name you chose for your's site bulletin is not listed first, you may mark it as the default by pressing the **Mark** button.

How To Configure TWB Bulletins

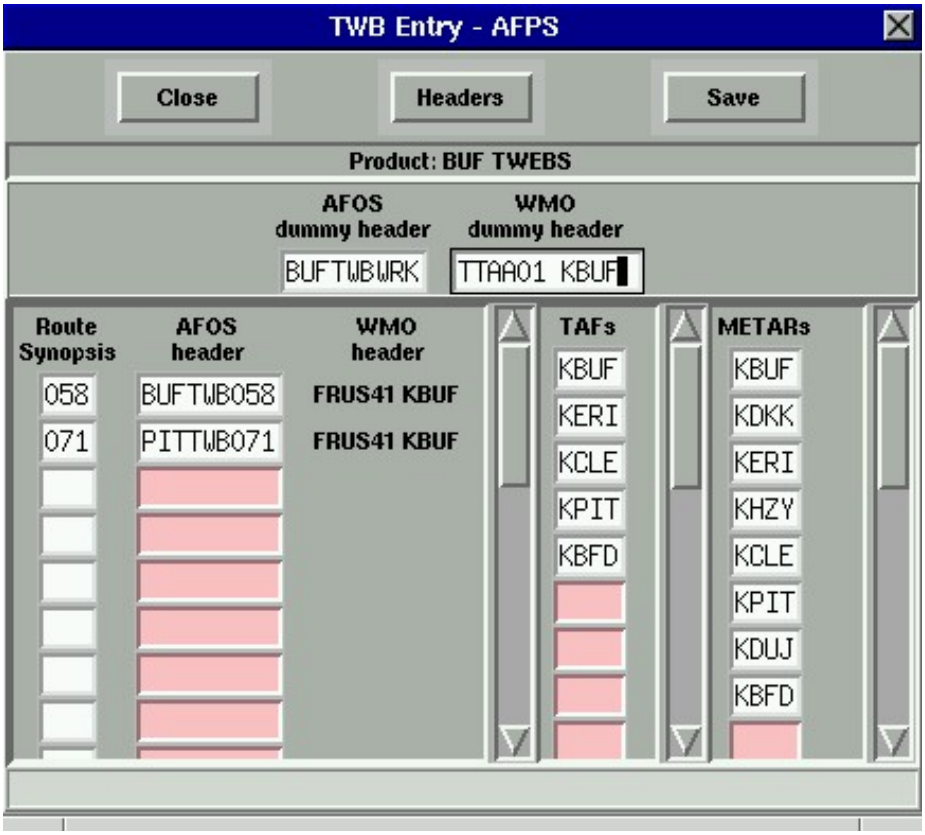
In normal circumstances, you should have one bulletin for all TWBs issued by your office and one for each office you may be asked to back up.

Select the **TWBs** button. A TWB setup dialog will be displayed



The dialog contains two sections. The top one, consisting of an entry field and **Add** button is used to add new bulletin definition. The bottom section, labelled **Configured TWBs** is used to view or modify existing bulletins.

Enter a name for your main bulletin in the entry field. This can be any sequence of letters and digits separated by blanks. Press the **Add** button. Another dialog will be shown.



Enter site ids for all sites you want to include in the bulletin. Leave the **AFOS header** column blank.
Press the **Headers** button. The program will scan file /awips/fxa/data/afos2awips.txt for TWBs matching site ids entered. The corresponding headers will be displayed. The TWBs will be transmitted with WMO headers as shown.

Though the AFOS header entries are editable, unless you are completely certain of what you are doing, you should not modify the values. The AWIPS transmission program, handleOUP.pl, will refuse to transmit forecasts if the AFOS header cannot be found in afos2awips.txt.
Note: Since the directory /awips/fxa is local, it is possible that the afos2awips.txt file on a workstation may differ from the one on the server. In such case you may run into trouble.

The two columns on the right, **TAFs** and **METARs** are for bulletins you want to view while preparing a TWB. Enter the 4−letter station id there.

The dummy headers are used to store work tafs in the text database. Those can be anything you want, provided that the standard format is followed, i.e. AFOS entry is 8 to 9 characters long.

Press the **Save** button. This action will first check whether the AFOS headers are valid. If not, an error dialog will show up and the procedure will abort. You will need to fix the headers (or the file afos2awips.txt).
If the headers are valid, several configuration files will be written to directory /awips/adapt/avnfps/etc/twbs. These are *yourname.def* and *yourname.xxx.list* where *yourname* is the bulletin name chosen (blanks replaced by underscores) and *xxx* is *twb*, *mtr* and *taf*. Also, skeleton template file *yourname.templ* will be created.

If there are no errors, you will see *Save succeeded* message in the status window (just above **Close** button). Press **Close** to close the dialog.
Note: You must enter both dummy headers for **Save** to succeed.

The bulletin label should show up in the **Configured TWBs** list. If you wish to modify the template, highlight the bulletin label and press the **Template** button. A text editor with loaded template file will be displayed. Make the changes and press **Save**, then **Close**.

The entries are separated by an = sign or by a blank line. The = sign will be stripped while loading into the editor. You may have an entry across several lines.

The **Verify** button can be used to check for existence of all required bulletin definition files.

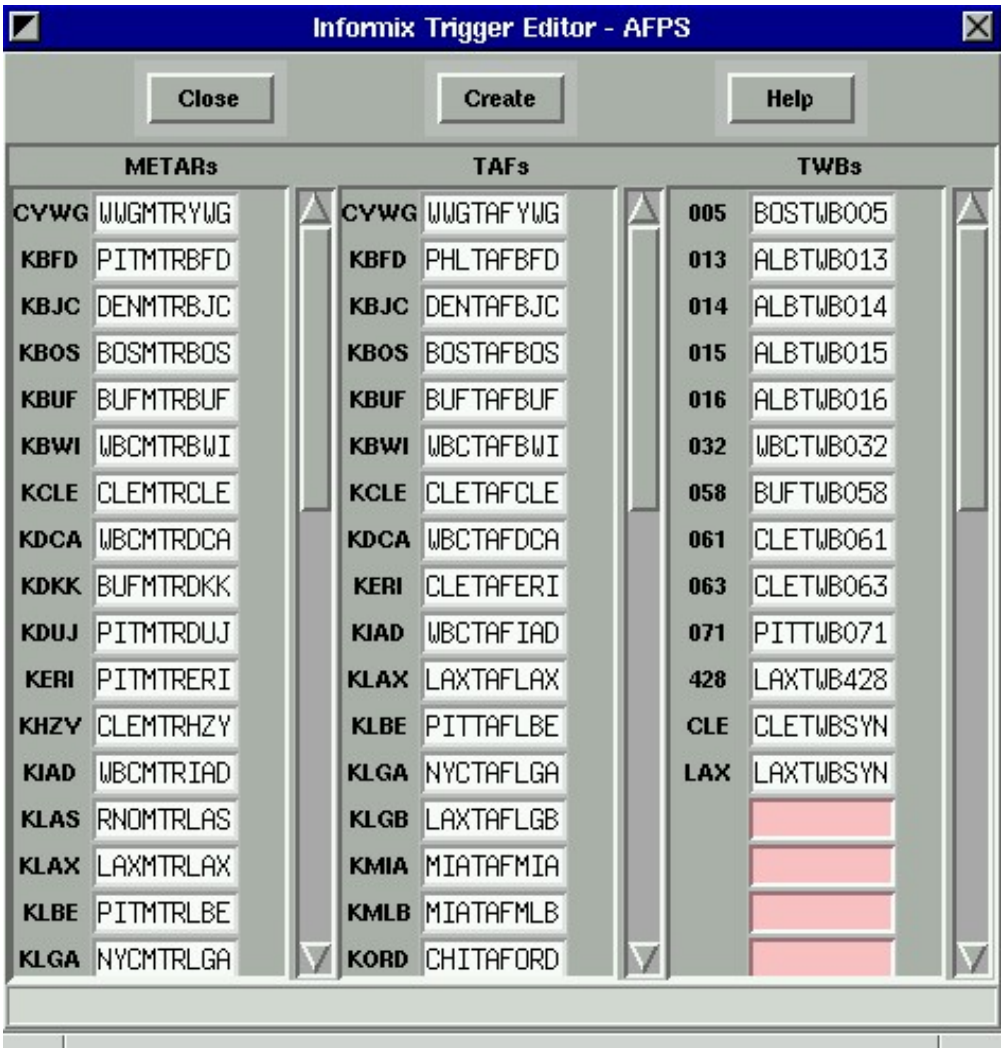
You may modify bulletin name. Highlight the old name, then press **Rename** button. This will pop up a dialog asking for the new name. Enter a name and press **OK**.

To delete unwanted bulletin, highlight it in the **Configured TWBs** list window, then press **Delete**. This will (requires confirmation) delete all files described above.

The bulletins are listed in alphabetical order. The sites listed in the first one are displayed in the **Load** dialog of the forecast editor. If the name you chose for your's site bulletin is not listed first, you may mark it as the default by pressing the **Mark** button.

How To Create Informix Triggers

You need to update the triggers each time you add/delete a bulletin, or add/delete station. Press **Triggers** button. Trigger entry form will appear.



The program collects and merges all the *.list files from all TAF and TWB bulletins and attempts to guess AFOS PILs for each of the bulletins types (TAFs, TWBs and METARs) and site ids. The TAFs and TWBs values are the same as those entered in the setup stage and most likely are correct. There is a possibility that the TAFs originating node (the CCC part of AFOS PIL) will be different than the corresponding METARs. You should check and modify, if neccessary all the values. Then press **Create** to write the trigger template file /awips/adapt/avnfps/etc/triggerTemplate This file needs to be processed by AWIPS localization script, mainScript.csh. Log on to the *data server* as user *fxa* and run the command

```
/awips/fxa/data/localization/scripts/mainScript.csh –trigger
```

This procedure will update the *watchwarn* table in *fxatext* database.

You may want to populate **avnfps** database to be able to start the monitoring GUI immediately. The utility runtrig.sh will extract most recent TAFs, TWBs and METARs from the text database. From the terminal window on the data server, enter

```
/awips/adapt/avnfps/bin/runtrig.sh –p [taf|twb] product ...
```

for all products that have been created.
Note: If the product name contains blanks you must replace them with underscores, for example, if the product name is *OKX backup*, you should type

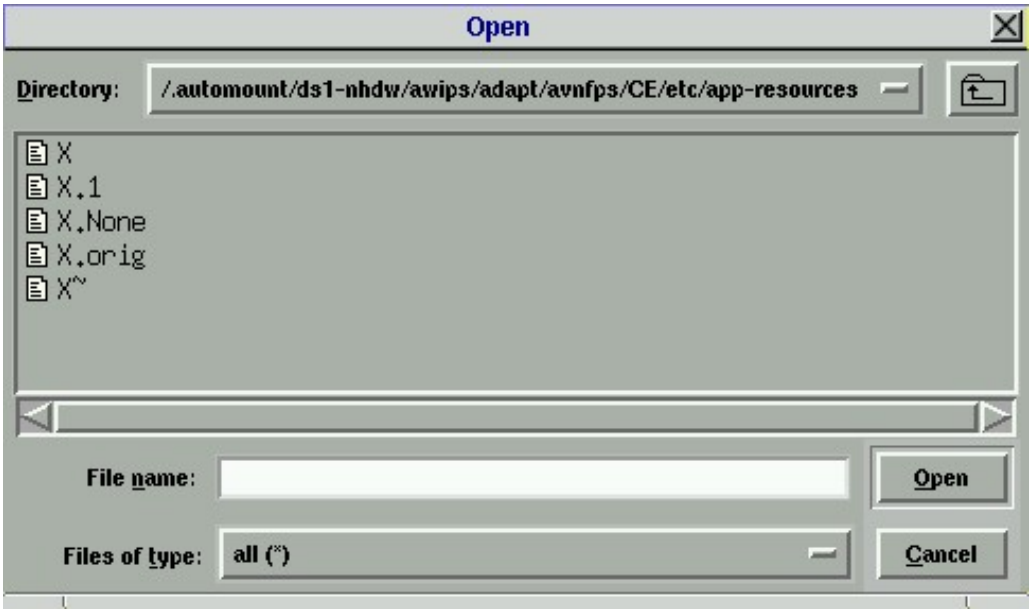
```
/awips/adapt/avnfps/bin/runtrig.sh -p taf OKX_backup
```

runtrig.sh will extract up to 6 METARs and 3 TAFs and TWBs (less if the text database contains duplicates).

How To Modify Resource Configuration File

You don't have to change resource configuration file(s). The two main programs, avnsetup.py and avnwatch.py use the default file: /awips/adapt/avnfps/etc/app−resources/X which is provided in the distribution package. Each forecaster can create his own resource file via resource configuration resource editor dialog available through the **Options/Setup** menu from the monitoring GUI. If you want to change the defaults for the whole office, in particular fonts and colors, we suggest to use the resource editor and set the resources there, then incorporate the changes into the default file.
Note: The default file contains resources **both** for avnsetup.py and avnwatch.py, the individual file contains only the resources needed by avnwatch.py.

Select the **X Resources** button. A file selection dialog will appear.



Select **X**, then press **Open** button. A text editor with loaded file /awips/adapt/avnfps/etc/app−resources/X will appear. This file contains default values and need not be modified. If you want to change values for specific resources follow the comments (i.e. lines starting with !!). The most likely values you may want to modify are:

```
!! used by avnsetup only
!! maximum number of sites in a collective forecast
!! those may need to be increased before attempting to run avnsetup
!! if you have a lot of TAFs
*TAFEntry*numrows:              12
*TWBEntry*numrows:              12
*Trigger*numrows:               30
```

The following option allows for changinng the way dialogs are treated by the window manager.

```
!! Set it to 0 if you want to iconify dialogs
*transientDialogs:              1
```

Transient dialogs behave as described in Overview. Non−transient dialogs are treated (almost) as stand−alone applications: window frame has the full set of options, including "Minimize" (i.e. iconify). You will also be able to change window stacking order and send dialog window to a different workspace. *Note*Beware of unexpected behavior. If a modal dialog (such as error message) is instantiated from some dialog and you chose to iconify the parent window, the whole application will appear stuck (you will see the watch icon over all displayed windows, no mouse/keyboard input will be processed).
Set **transientDialogs:* to 0 if you want non−transient dialogs. All *modal*dialogs are always transient.

If you want to see a warning dialog when closing dialogs, change the value of *confirmClose* to 1.

```
!! confirmation on closing dialogs
*confirmClose:                  1
```

If you have external speakers connected (the witch's laugh is not sufficienly loud for the internal speaker), you may want to uncomment/change the **notifyPlay:* and **playFile* lines:

```
!! *notifyPlay:                 pale green
*playFile:                      /data/local/avn/etc/sounds/asterisk.au
```

The playFile can be any sound sample you like.
Other alert thresholds are:

```
*notifyDeiconify:               yellow
*notifyRaise:                   yellow
```

You may always change the values from the main menu in the GUI.

```
!! time between consecutive checks in seconds
*timeout:                       30
```

The line:

```
*disallowSend:                  Error
```

requires the forecaster to clear the orange and red errors before forecast can be transmitted (menu **File/Clear**). Depending on the office policy, the value can be set to *Warning* or *Error*.

```
!! use template, one of: Template, Merge, Previous
*loadOrder:                     Merge
```

Again, you can always select the order in the load dialog.

```
!! update TAF with current METAR while loading previous forecast
*usemetar:                      1
```

If you don't like this, set it to 0. This setting also can be changed from the **Options** menu in the main GUI.

```
!! change 120 to 140 if you want bigger default font
*font:              −adobe−helvetica−bold−r−normal−*−*−120−*−*−p−*−iso8859−9
```

To display METARs with the forecast editor, similarly to the Aviation Workstation, set the following option to **vertical**. Other valid entries are **horizontal**: the METAR window is to the right of the editor, or **none**: METARs are not shown.

```
!! horizontal, vertical or none
*ForecastEditor*orientation:    vertical

!! maximum number of sites in a collective forecast
!! those may need to be increased before attempting to run avnsetup if you have a lot of TAFs
*TAFEntry*numrows:              12
*TWBEntry*numrows:              12
*Trigger*numrows:               24
```

How To Find Color Names

Valid color names are listed in the file /usr/lib/X11/rgb.txt. You can specify either color name (which can be one or two words) or the RGB values in hex notation preceeded by a #.
The resource editor has a built−in color editor.

How To Find Font Names

The font name structure is a complicated one. It consists of several fields separated by a "−" sign, see the **font* line above. The three important fields are:

- 3rd: font weight (normal, bold, etc)
- 8th: font size
- 11th either p for proportional or m for monospace

In addition, font names can be aliases, such as 10x20 above.
The resource editor has a built−in font viewer.

Forecasters may override resources specified in the default file. avnwatch.py attepts first to open /awips/adapt/avnfps/etc/app−resources/X.n, where *n* is the forecaster number. If the file does not exist, the default file /awips/adapt/avnfps/etc/app−resources/X is read.

The forecaster specific file can be edited using resource editor dialog available from **Options** menu in the monitoring GUI.

How To Customize Monitoring Rules

avnfps allows you to customize TAF alert criteria for individual airports. There are several (currently 13) built−in rules. These are:

DDDelta
: Category: *wnd*
Activated when: TAF and METAR winds directions differ by *DD* with forecasted or observed wind speed >= *FF*.
Arguments: *DD FF*
Example values: `DD = 30, FF = 12`

FFDelta
: Category: *wnd*
Activated when: TAF and METAR wind speeds/gusts differ by *FF* with forecasted or observed wind speed or gust >=*FF1*
Arguments: *FF FF1*
Example values: `FF = 10, FF1 = 12`

XFFMetar

Category: *wnd*
Activated when: METAR runway cross wind speed >= *FF* kt
Arguments: *FF*
Example values: `ff = 10`

CigCatDelta

Category: *cig*
Activated when: TAF and METAR ceiling differ by *#categories*,
Arguments: *#categories*
Example values: `#categories = 1`

VsbyCatDelta

Category: *vis*
Activated when: TAF and METAR visibilities differ by *#categories*,
Arguments: *#categories*
Example values: `#categories = 1`

CigTafThreshold

Category: *cig*
Activated when: TAF ceiling <= *CIG1* and METAR ceiling > *CIG2* or unlimited
Arguments: *CIG1 CIG2*
Example values: `CIG1 = CIG2 = 3000`

CigMetarThreshold

Category: *cig*
Activated when: METAR ceiling <= *CIG1* and TAF ceiling > *CIG2* or unlimited
Arguments: *CIG1 CIG2*
Example values: `CIG1 = CIG2 = 3000`

VsbyTafThreshold

Category: *vis*
Activated when: TAF visibility <= *VSBY1* and METAR visibility > *VSBY2*
Arguments: *VSBY1 VSBY2*
Example values: `VSBY1 = VSBY2 = 0.5`

VsbyMetarThreshold

Category: *vis*
Activated when: METAR visibility <= *VSBY1* and TAF visibility > *VSBY2*
Arguments: *VSBY1 VSBY2*
Example values: `VSBY1 = VSBY2 = 0.5`

WxTafDelta

Category: *wx*
Activated when: *WX1* or *WX2* or *WX3* occurs in in TAF and not in METAR.
Arguments: *WX1 WX2 WX3* (some may be empty)
Example values: `WX1 = 'TS', WX2 = WX3 = ''`

WxMetarDelta

Category: *wx*
Activated when: *WX1* or *WX2* or *WX3* occurs in in METAR and not in TAF.
Arguments: *WX1 WX2 WX3* (some may be empty)
Example values: `WX1 = 'TS', WX2 = WX3 = ''`

WxVsbyDelta

Category: *wx*
Activated when: *WX1* or *WX2* occurs in in METAR while not in TAF, or *WX1* or *WX2* occurs in TAF while not in METAR with visibility <= *VSBY*
Arguments: *WX1 WX2 VSBY*
Example values: `WX1 = 'RA', WX2 = 'SN', VSBY = 5.0`

Each of these rules accepts a message to be displayed when the rule "fires" and one or more arguments that you can set in the rule editor. For each set of arguments, the rule has associated color (severity level). The following colors are used, in order of increasing severity:

- *Green* − no warning
- *Grey* − missing data
- *Pale Green*
- *Yellow*
- *Orange*
- *Red*
- *Purple*

Each rule belongs to one of the four categories (corresponding to labels in the status window of the monitoring GUI):

- *wx* − weather
- *vsby* − visibility
- *wnd* − wind
- *cig* − ceiling

The monitoring program **avnwatch.py** checks current weather conditions against each rule on the list. If the rule is activated, the label will get its balloon message and background color set to those configured for the rule.
If more than one rule is activated for a category, the background color is set to the highest severity.
Some of the rules are unique: only the message corresponding to the highest severity will be shown. These are: *DDDelta*, *FFDelta*, *CigCatDelta*, *VsbyCatDelta* and *WxVsbyDelta*. This is intended to avoid display such as:

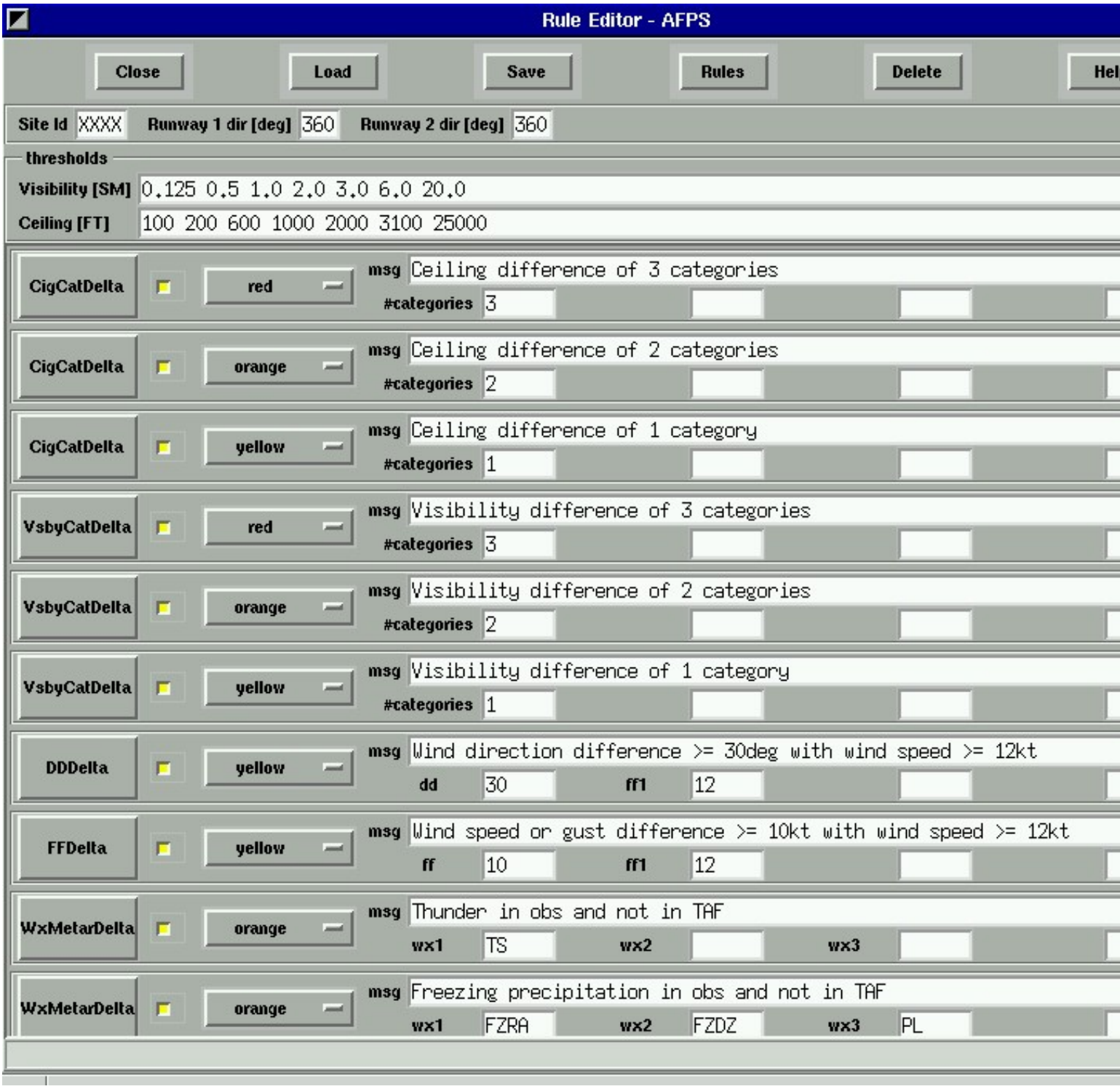
- `Visibility difference of 3 categories`
- `Visibility difference of 2 categories`
- `Visibility difference of 1 category`

For example, the rule *WxMetarDelta* checks for occurence of weather condition(s) present in METAR but not forecast. The rule accepts up to three arguments: weather type(s). The monitoring criteria provided with the package uses this rule twice:

1. color: yellow
   message: Thunder in obs and not in TAF
   wx1: TS, wx2: blank
   This rule will "fire" when TS is in recent METAR, but not in TAF, regardless of visibility. In such case the wx field in the monitoring GUI will have yellow background
2. color: yellow
   message: Freezing precipitation in obs and not in TAF
   wx1: FZRA, wx2: FZDZ
   This rule will "fire" when freezing rain or drizzle is reported, but not present in TAF The wx field in the monitoring GUI will have orange background

The rules for each site are stored in a file CCCC.rules where CCCC is the site id. If this file does not exist, the default rules defined in the file XXXX.rules are used. As delivered this file provides minimum U.S. TAF amendment criteria, as described in NWSI 10−813.

If you choose to modify the rules, select **Watch Rules** from the main GUI.



The rule editor will be displayed, with the site id set to *XXXX*. If you only want to modify rules for one site, replace *XXXX* by the real site id, then press **Load**.

Apart from the rules, the following values can be modified:

Runway N dir [deg]

If you want to use runway cross wind as one of your rules, enter runway direction(s) here. Value −1 means missing.

Visibility [SM]

This field accepts a sequence (separated by blanks) of real numbers. The values will be plotted as a reference grid on the weather plot dialogs activated from the monitoring (avnwatch) program. The first and last value resticts the range of values to be plotted. The remaining values are used to set visibility categories for the **VisCatDelta** rule.

Ceiling [FT]

This field accepts a sequence (separated by blanks) of integer numbers. The values will be plotted as a reference grid on the weather plot dialogs activated from the monitoring (avnwatch) program. The first and last value resticts the range of values to be plotted. The remaining values are used to set ceiling categories for the **CeilCatDelta** rule.

The categories for ceiling and visibility are defined as the smallest index for an element on the ceiling/visibility list with value greater than the observation/forecast ceiling/visibility. For example, given the default list of visibilities:

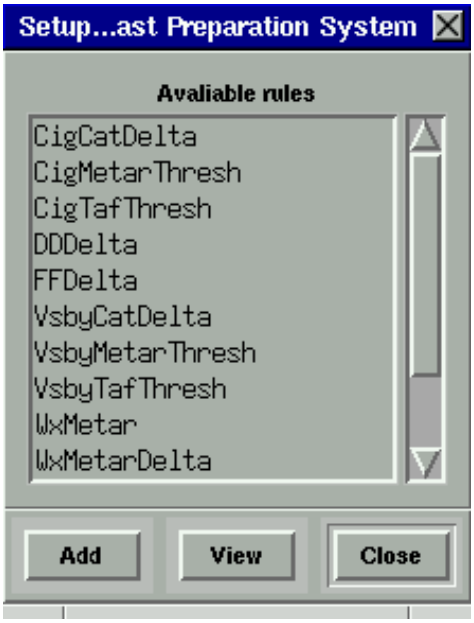
```
0.5 1.0 2.0 3.0 6.0
```

visibility 1/4SM corersponds to category 0, 2SM is category 4, 6SM and more gives category 5.

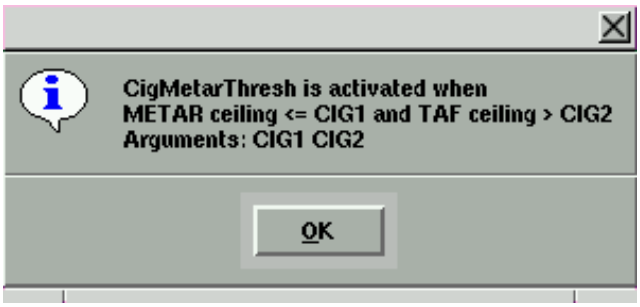
If you are happy with the rule selection and only want to modify the arguments, modify the values in entry fields.
You can change rule severity via the option menu.

If you want to delete an existing rule, select it by pressing the button with rule name (on the left), then press **Delete**.

The toggle button can be used to deactivate the rule, without deleting it. To add a new rule, press **Rules** button. A dialog containing list of available rules will appear.



You can view a rule by selecting it on the list, then pressing **View** button.



This is the same message that will appear as a balloon associated with the rule button in the rule editor.

Press the **Add** button. The selected rule will be inserted before a rule that is currently selected in the rule editor (there is always one, note the blank rule at the end of the list). Some default arguments will be set, which you probably want to modify.

When finished with the modification, press **Save**. The program will check entered values for proper format. In case of a string entered where an integer value is expected, an error dialog will appear.
Note. The program will not check whether the values are meaningful, such as weather element, or, if the lower limit is indeed less than the upper.

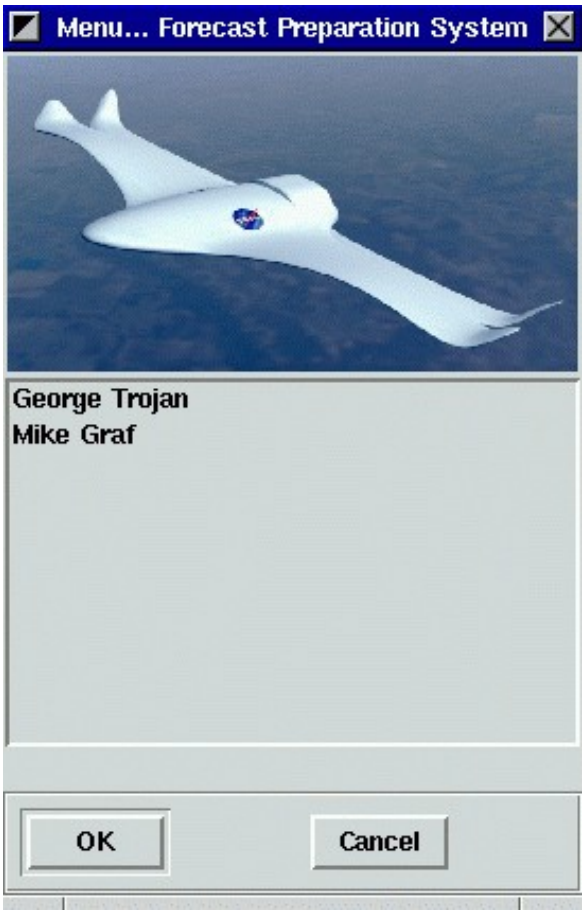
Press **Close** to close the dialog.

AVNFPS TAF Monitor

Release 1.0
June 19, 2003

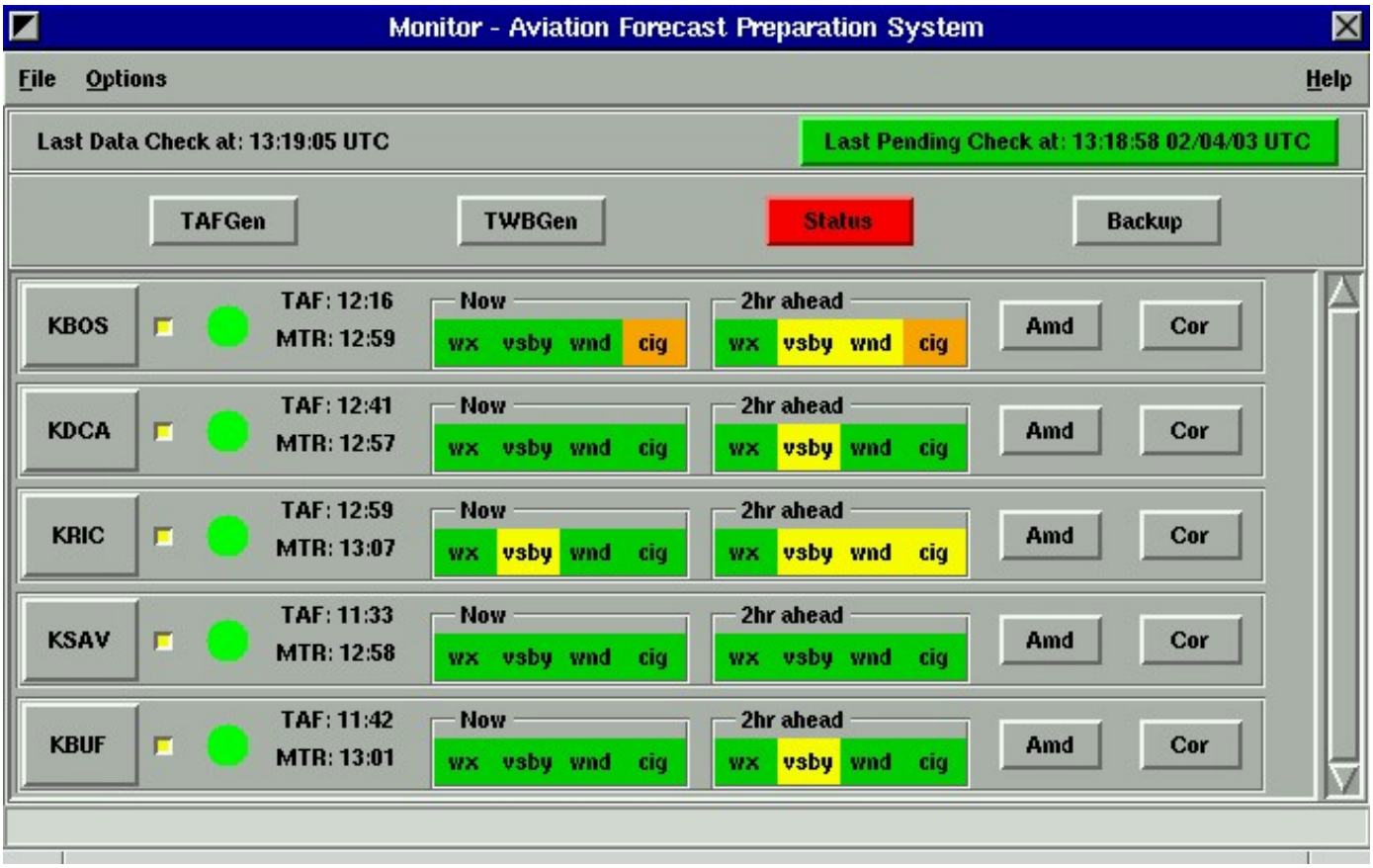
How To Start avnwatch

The program is normally started from window manager menu. The details depend on the window manager: pop−up menu in dtwm (HP−UX), panel menu or a short−cut icon in Sawfish (Linux). In any case, the avnwatch.sh script is executed. The script calls avnmenu.py.



The GUI displays list of forecasters. You can select your name from the list, in which case your own X resources file will be read on the startup of avnwatch.py. If you don't have such file, just press **OK**. Forecaster name can be changed later within the forecast editor window.

After 2 seconds the menu goes away and the TAF monitoring GUI should appear (it may take much longer on a busy HP workstation). The GUI will monitor TAFs from the default bulletin, as specified by the setup program. The program will monitor all the sites listed in the bulletin(s). You can always change the selection through **File/Product** menu.

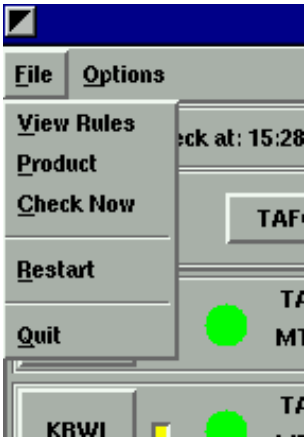


GUI Description

The GUI contains several areas: menu bar, status bar, command buttons, site status display window.

Menu Bar

- **File**



View Rules

Displays site information and rules used to monitor forecasts for selected site. This is the same dialog that is used by the setup program. When started from this GUI, the Rule Editor dialog allows only to

activate/deactivate currently defined rules (you can not add/delete a rule or change its parameters). Reload currently monitored product(s). After making changes, press the **Save** button in the dialog, then reload currently monitored product via the **File/Product** menu.

Product

Invokes list dialog allowing selection of bulletins to monitor. You can select one or more bulletins. You should reload current product if you modified monitoring rules.

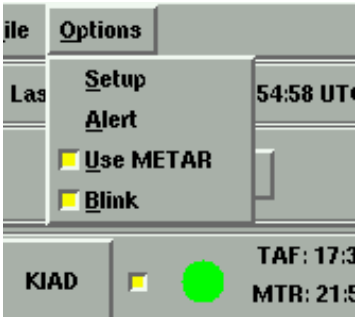
Check Now

Normally the checks are done every 30 seconds. This option forces an immediate check of all TAFs versus current observations and the transmission status. An alert, if warranted, will be raised.

Quit

Terminates the application. If the *confirmClose option is set in the X resources file, you will get confirmation dialog which may save you retyping several TAFs. This action is equivalent to selecting **Close** from window manager menu.

- **Options**



Setup

Can be used to edit forecaster specific configuration file /awips/adapt/avnfps/etc/app−resources/X.n. See resource editor below.

Alert

Used to select alert criteria when the monitor detects a condition requring forecaster's action. Alert pops up a dialog in which you may select severity level for activating particular option.

Use METAR

If selected, the TAFs loaded when the editor is invoked, will be updated with the current observation, provided that the current METAR time is not older than 30 min

Blink

If selected, the color circle label will blink on new notification about changing weather conditions.

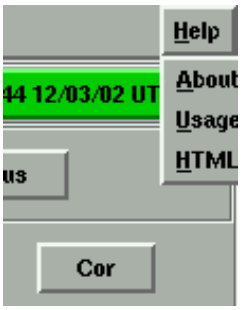
Deiconify
When the main GUI is iconified, it will deiconify self

Raise
When the main GUI is buried under other windows, it will pop up to the top

Play
Plays an audio file (specified in X−resources).

Voice
Appropriate voice message will sound. Not yet available with this release.

- **Help**



About
Provides verion number, and contact information

Usage
Displays help window

HTML
Displays this document with the selected browser (most likely Netscape)

Status Bar

The **Last Data Check** label displays the most recent time observations and TAFs were checked. The frequency of checks is set in the X resources file (item *timeout). The **Last Pending Check** button displays the last time the transmission program **avnxmitserv.py** checked the transmission directory for pending forecasts. The background color corresponds to the time interval between last checked value and last run of **avnxmitserv.py**:

- less than 1 min: green
- between 1 and 2 min: orange
- more than 2 min: red

If the difference is greater than 24 hours, the time displayed is 00:00:00 UTC.

Command Buttons

TAFGen

This button invokes forecast editor initialized to TAFs. No forecasts are loaded.

TWBGen

This button invokes forecast editor initialized to TWBs. No forecasts are loaded.

Status

This button invokes monitoring dialog displaying data reception and forecast transmission log files. The dialog can also be used to retransmit forecasts.

The background color is used to express status of the last transmission attempt. Three colors are used:

◊ *Grey*: The transmission server did not try to transmit bulletins

◊ *Green*: All files in /awips/adapt/avnfps/xmit/pending/ were transmitted successfuly

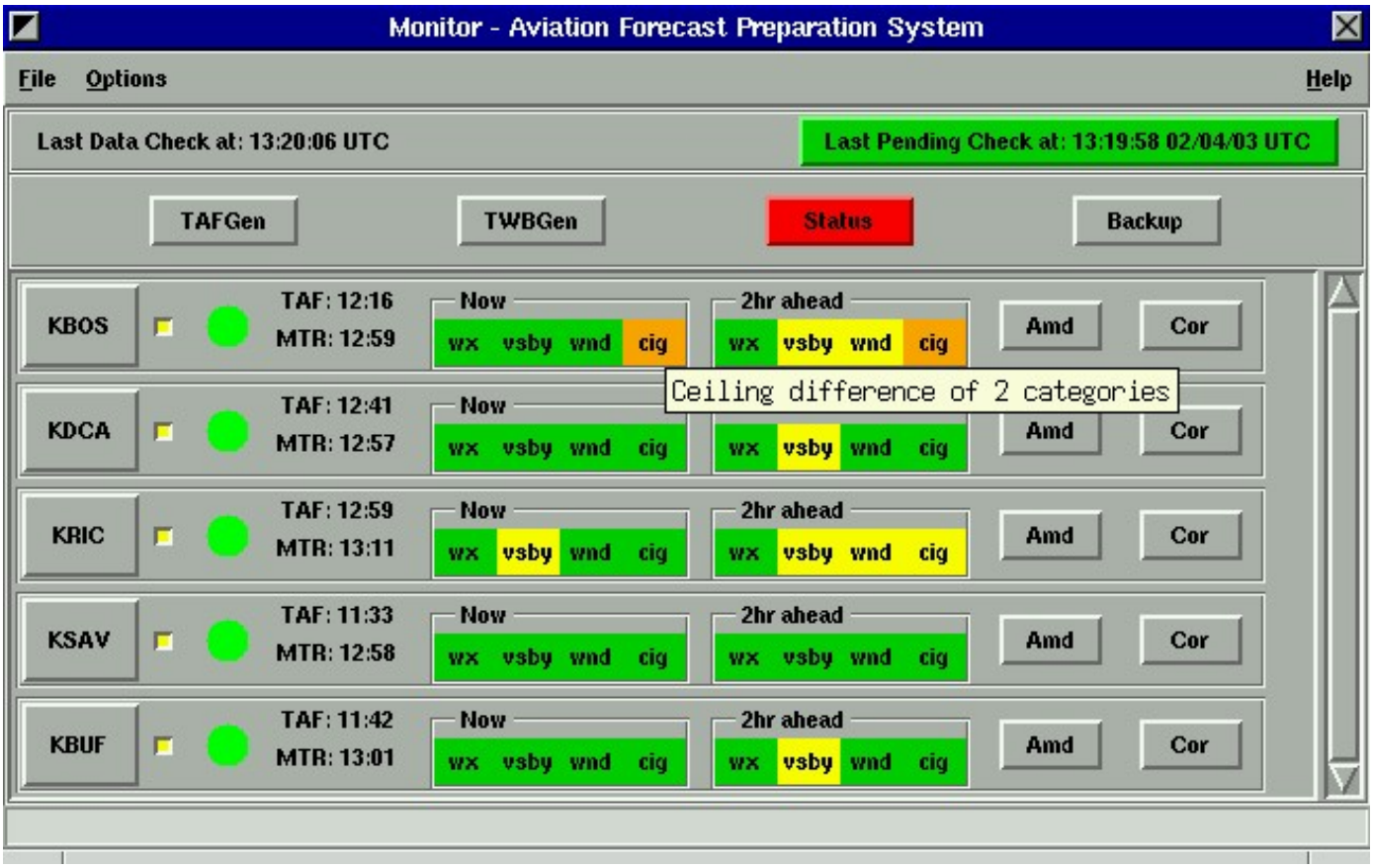
◊ *Red*: At least one of the transmission attempts failed

Site Status Display Window

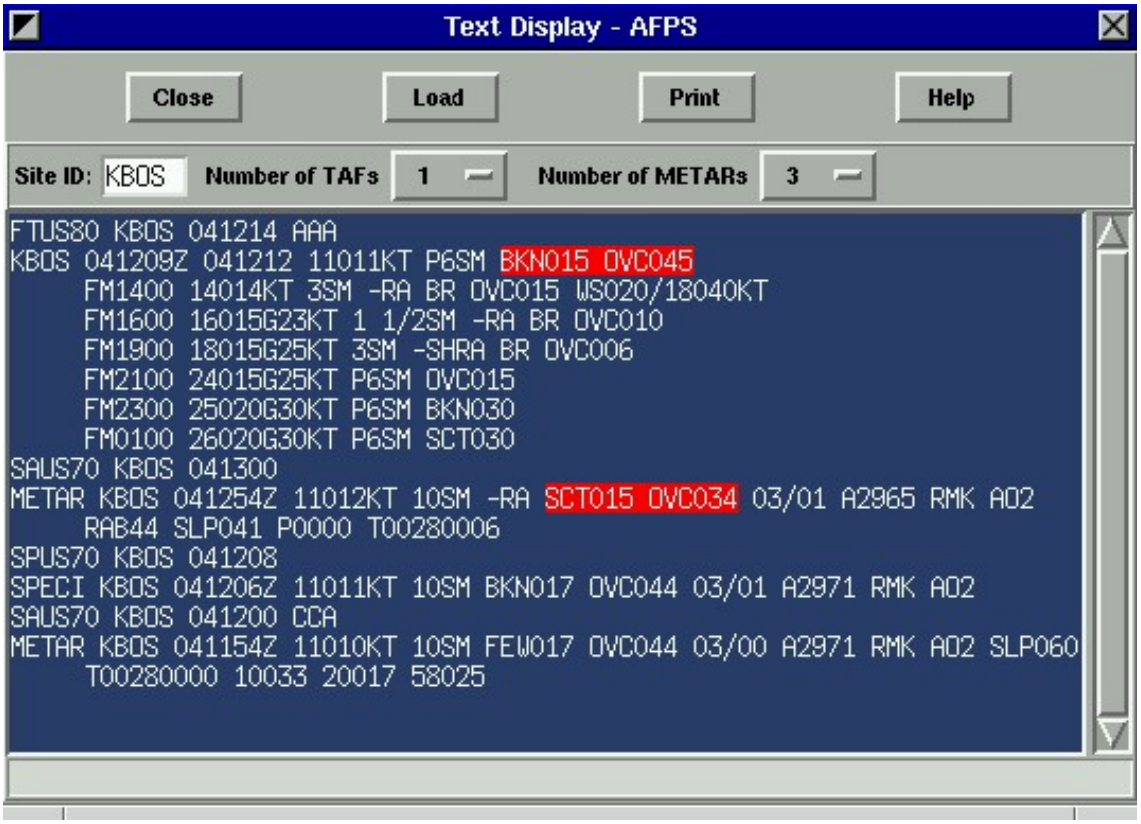
The monitoring window displays list of site−specific information areas. Each area consists of the following units:

- Site id button. This button invokes text product dialog displaying recent TAFs and METARs as text. If the TAF fails quality control check, a warning dialog wil be shown.
- Active toggle button. The site is monitored only when the button is 'on'. The usefulness of this option has been questioned and the button can be removed in the future.
- Label used to notify about possible problems with the data. The background color is used to distinguish between the following:
  Missing report or a programming bug: red
  The TAF decoder found problems with the TAF: orange. Normally, this should not happen − all the TAFs should pass QC before transmission
  The forecasted weather in TEMPO or probability groups has not materialized within the last 2 hours: yellow
  The coresponding message is shown in the balloon popup window. If none of the above occurs, the label's backgroungd will be grey, making it invisible.
- Last TAF and METAR time labels: those display time when the forecast/obs where written uder the **avnfps** data directory. The background color is used to notify the forecaster when the report is late. When the last received report is older than threshold value, the background becomes yellow. The threshold value is 1hr 5min for METARs and 6hr 40 min for TAFs.
  The TAF time is the time of the most recent *valid* forecast.
- Current TAF status: contains four labels, one for each forecast element: weather, visibility, wind and ceiling. There are color−coded, green if the element passed all the criteria, pale green, yellow, orange, red and purple, depending on the severity of the violated rule(s) otherwise. When you point at the label, a balloon popup window will display messages for all violated rules.
  Current weather is compared to TAF valid at the current time (rounded down to the nearest hour, with an exception for amended/corrected TAF, with valid period starting within the next half hour).
- 2 hr TAF status: same as the above, but current weather conditions are compared to the forecast 2 hours ahead.
- **Amd** Invokes TAF editor initialized for amended TAF.
- **Cor** Invokes TAF editor initialized for corrected TAF.

If you point mouse cursor in one of the labels of the status windows, a balloon message will show the conditions that are associated with the relevant warning, or **OK** if none of the monitoring rules was violated.



Text Product Display Window



When invoked, the dialog displays the most recent TAF and 3 METARs for the selected site. The number of TAFs to display can be changed from the menu: **Number of TAFs** can be set to 1, 2, 3 or 99 (last 24 hours). The number of METARs can be adjusted from **Number of METARs** menu: 1, 2, 3, 6, 12 or 99 (last 24 hours). Once the dialog is up, it is possible to change the site id through the **Site ID** entry field (the site does not have to be currently monitored).
Note: The dialog does not update its contets as new data is available. Use the **Load** button to reload most recent METARs and *valid* forecasts.

There are three command buttons in this dialog

Load
: Reloads data for the current selection

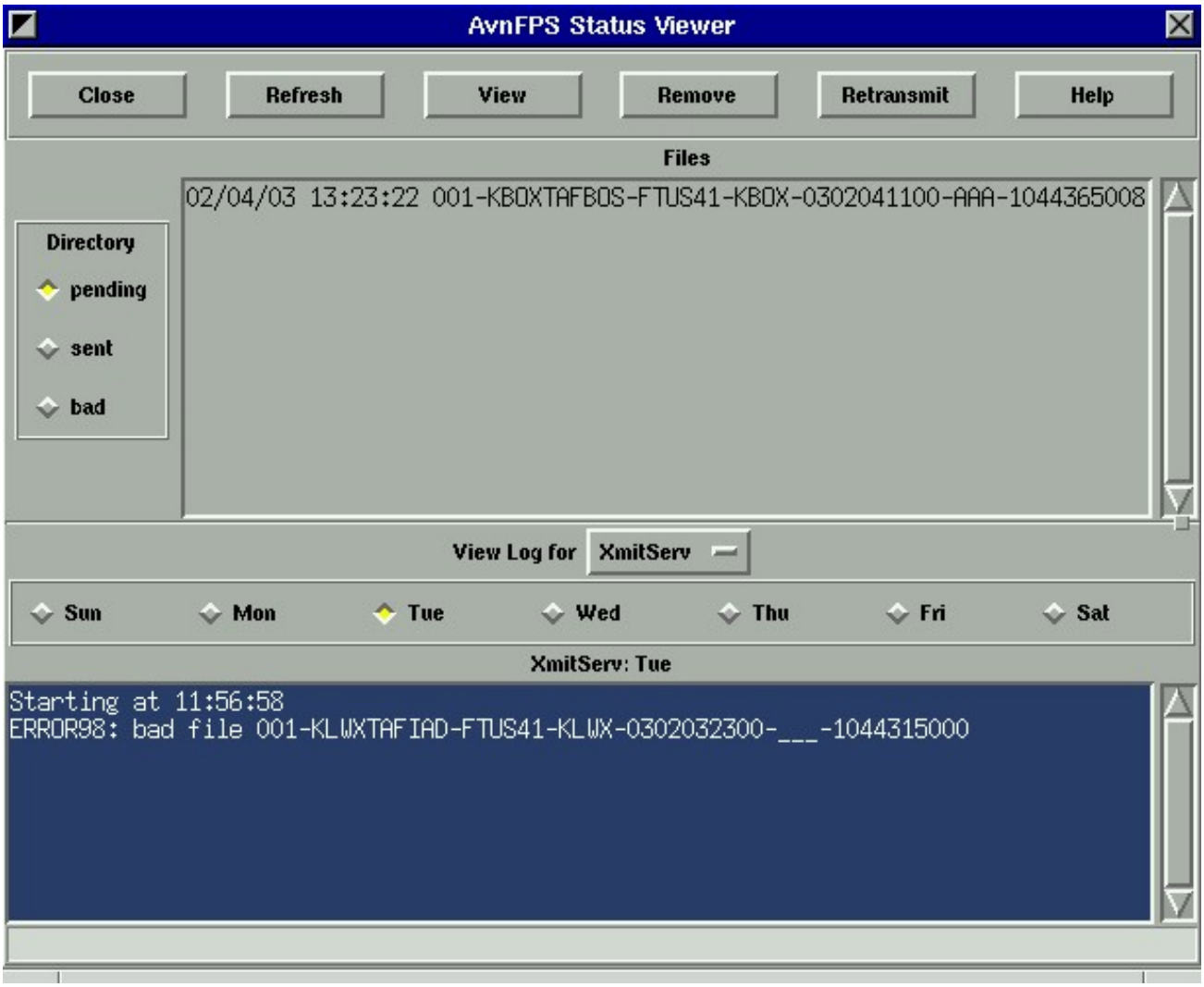
Print
: Invokes the print dialog. The dialog displays print command: lp for HP−UX, lpr or lp for Linux systems. If you want to redirect output to a particular printer, enter appropriate options (for example: lp −dlp2). You can use this dialog to dump content of the text window to a file: enter cat > filename to write to filename.

Help
: Displays the help dialog

The **Options** menu allows for quick change of fonts and colors. The default values are set in the X resources file.

When there is an alert for a given site, the weather element(s) that caused the alert are highlighted within the forecast and the most recent METAR.

Status Monitor Window



This dialog is used to view log files from triggers, decoders and the transmission server, as well as queue of forecasts waiting to be transmitted. The dialog consits of two areas:

- The top part is for viewing the content of directories: /awips/adapt/avnfps/xmit/pending/ (default): /awips/adapt/avnfps/xmit/sent/ and /awips/adapt/avnfps/xmit/bad/. Directory selection is done through radiobuttons on the left.
- The bottom part displays content of the log files for the three trigger programs: mtrtrigger, taftrigger, twbtrigger and the transmission server avnxmitserv.py. The **View Log for** option menu selects the log file. The row of radiobuttons below the menu can be used to view the log file for a specific day. By default, the log files for the current day are displayed. See Logging section in Overview for file format.

The buttons on the top of the GUI serve the following purpose:

Refresh
: Forces the dialog to rescan transmission directories and reread the log files. The dialog does not refresh its content periodically.

View
: Displays content of a selected file from the transmission directory

Remove

Removes selected files from the transmission directory. The intended use is to prevent transmission of bad forecasts in case of rapidly changing weather.

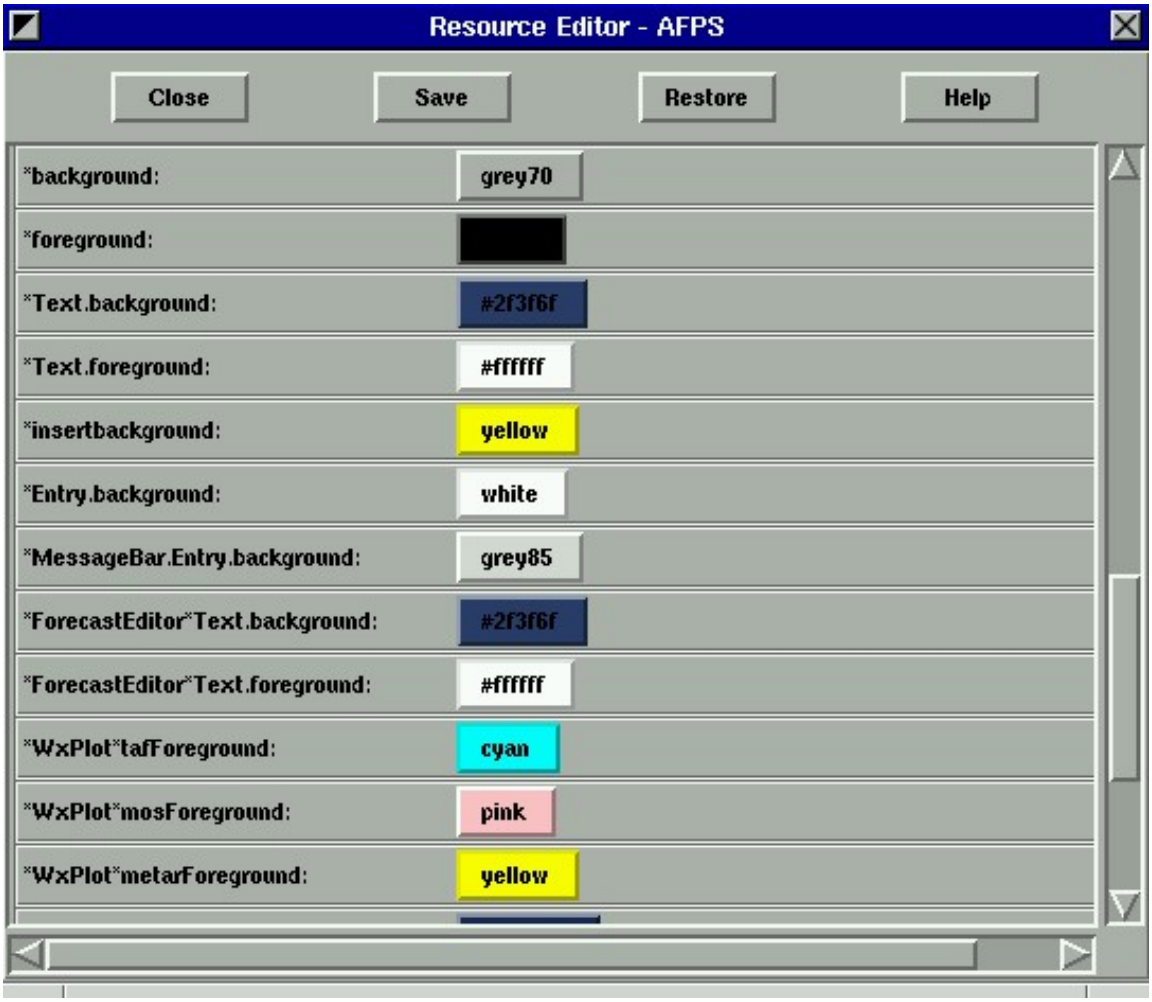
Retransmit

The selected file is moved (if it is not already in) to /awips/adapt/avnfps/xmit/pending/ and has an **!** mark appended to the name. This tells the transmission server to transmit the file regardless of the transmission window. The forecast header and issue times are not updated.

Help

Displays the help dialog

Resource Editor Window



This dialog can be used to edit resource configuration files for individual forecasters. The default file *has* to be maintained through a text editor. Within the scrolled window there is a list of configuration resources. Each frame contains name of the resource, you can display a short description in a balloon popup window by pointing mouse cursor at the name. Value of the resource is displayed on the right. Depending on the resource type it is:

Toggle Button

Used for Boolean values: yes or no (1 or 0 in the configuration file)

Option menu

Allows selection from predefined values

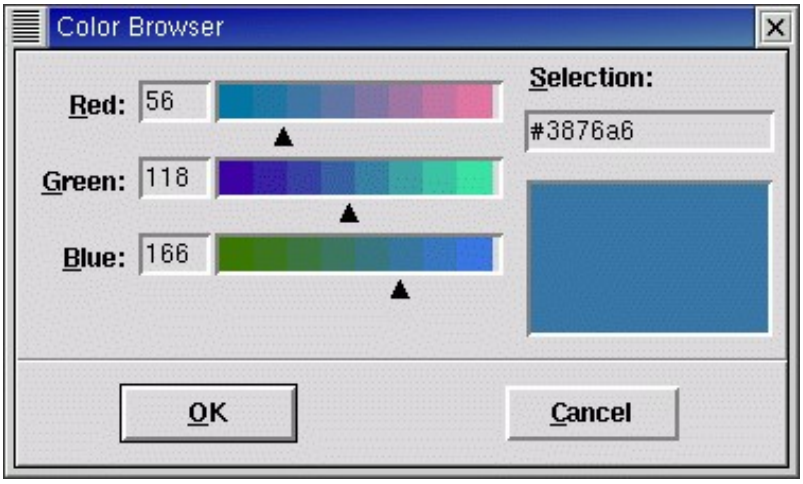
Entry

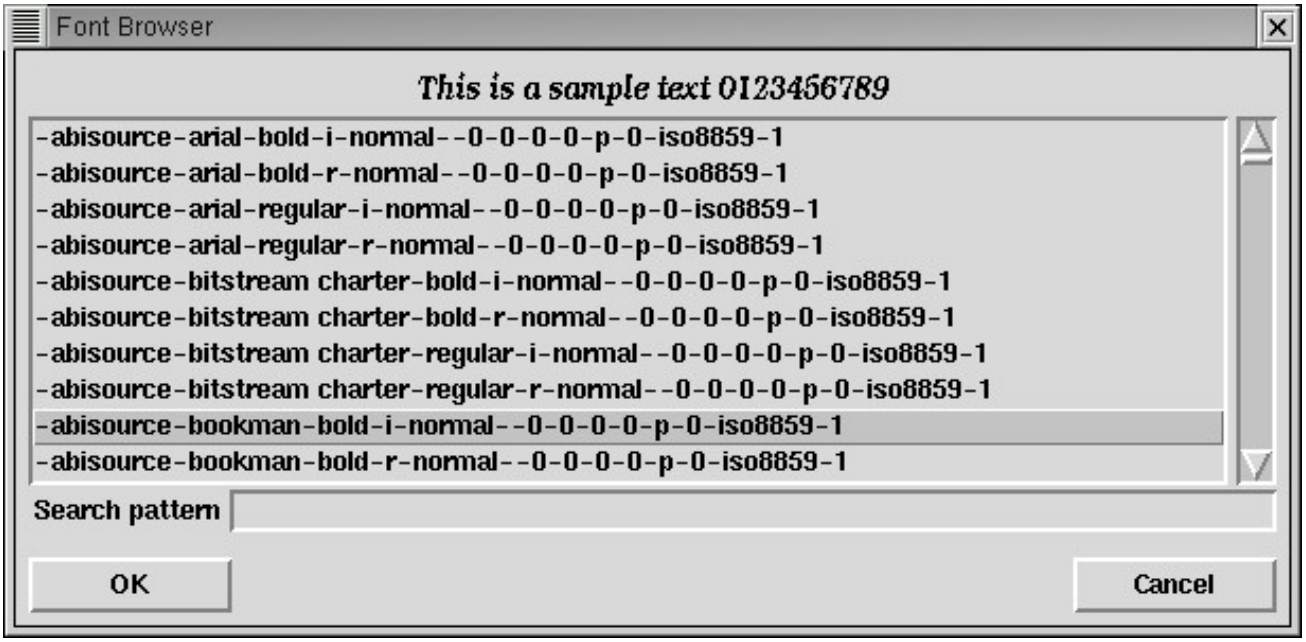
Allows to type in values, such as width and height

Button

Invokes dialogs specific for a given resource. This can be

- ◊ **File selection dialog:** to select audio file, resource *playFile
- ◊ **Color chooser dialog:** to define colors
- ◊ **Font chooser dialog:** to define fonts





When the editor starts, it tries to access configuration file awips/adapt/avnfps/etc/app−resources/X.n where n is forecaster number. If the file does not exist, a warning dialog is displayed and the default file is read.

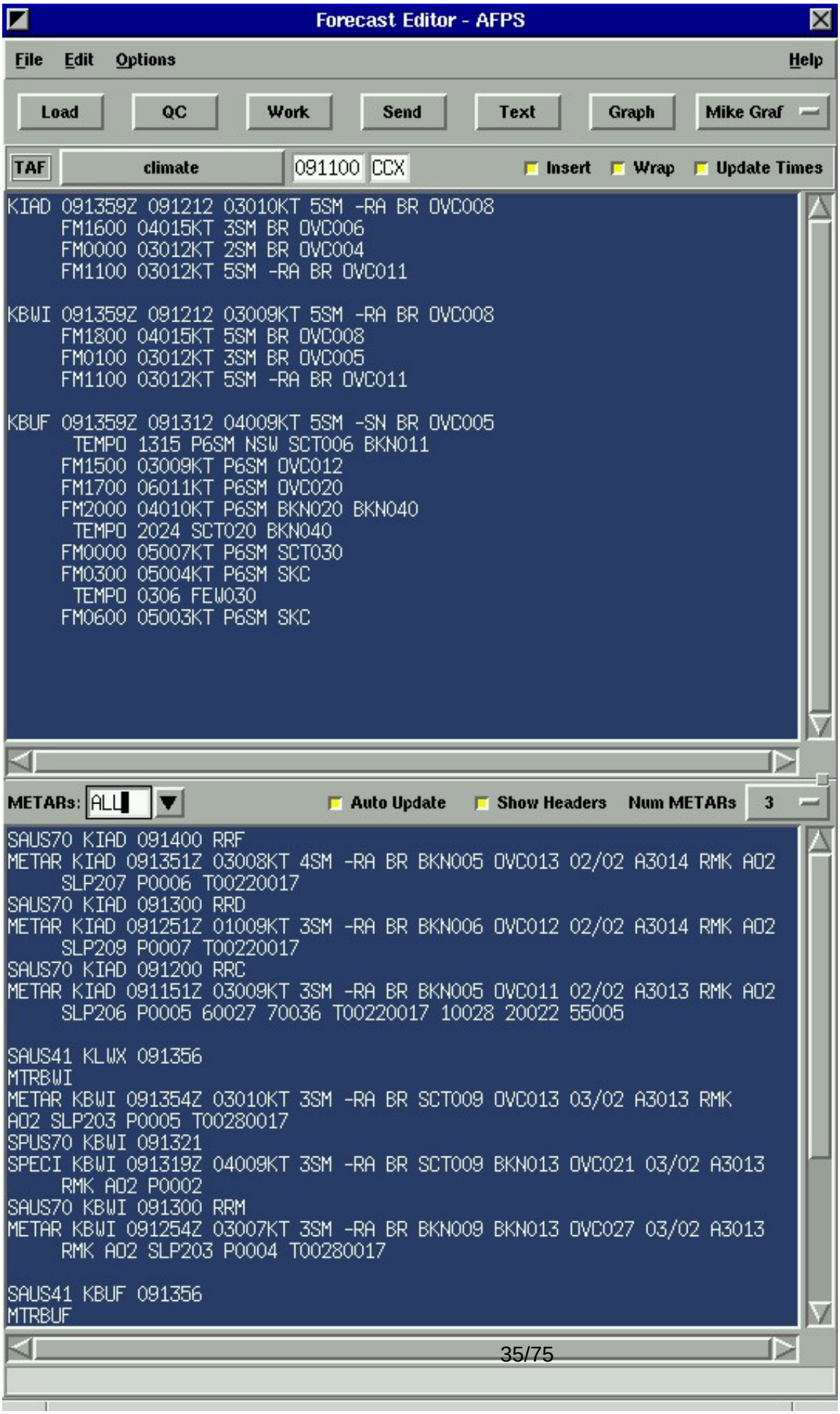
To save modified resources, press **Save** button. This will write the configuration file, stripping all comments. You will still be able to edit the file using text editor of your choice.

You may delete your configuration and start again from the defaults by pressing **Restore**.

AVNFPS Forecast Editor

Release 1.0
June 19, 2003

GUI Description


Forecast Editor - AFPS
File Edit Options Help
Load QC Work Send Text Graph Mike Graf
TAF climate 091100 CCX Insert Wrap Update Times
KIAD 091359Z 091212 03010KT 5SM -RA BR OVC008
FM1600 04015KT 3SM BR OVC006
FM0000 03012KT 2SM BR OVC004
FM1100 03012KT 5SM -RA BR OVC011
KBWI 091359Z 091212 03009KT 5SM -RA BR OVC008
FM1800 04015KT 5SM BR OVC008
FM0100 03012KT 3SM BR OVC005
FM1100 03012KT 5SM -RA BR OVC011
KBUF 091359Z 091312 04009KT 5SM -SN BR OVC005
TEMPO 1315 P6SM NSW SCT006 BKN011
FM1500 03009KT P6SM OVC012
FM1700 06011KT P6SM OVC020
FM2000 04010KT P6SM BKN020 BKN040
TEMPO 2024 SCT020 BKN040
FM0000 05007KT P6SM SCT030
FM0300 05004KT P6SM SKC
TEMPO 0306 FEW030
FM0600 05003KT P6SM SKC
METARs: ALL Auto Update Show Headers Num METARs 3
SAUS70 KIAD 091400 RRF
METAR KIAD 091351Z 03008KT 4SM -RA BR BKN005 OVC013 02/02 A3014 RMK AO2
SLP207 P0006 T00220017
SAUS70 KIAD 091300 RRD
METAR KIAD 091251Z 01009KT 3SM -RA BR BKN006 OVC012 02/02 A3014 RMK AO2
SLP209 P0007 T00220017
SAUS70 KIAD 091200 RRC
METAR KIAD 091151Z 03009KT 3SM -RA BR BKN005 OVC011 02/02 A3013 RMK AO2
SLP206 P0005 60027 70036 T00220017 10028 20022 55005
SAUS41 KLWX 091356
MTRBWI
METAR KBWI 091354Z 03010KT 3SM -RA BR SCT009 OVC013 03/02 A3013 RMK
AO2 SLP203 P0005 T00280017
SPUS70 KBWI 091321
SPECI KBWI 091319Z 04009KT 3SM -RA BR SCT009 BKN013 OVC021 03/02 A3013
RMK AO2 P0002
SAUS70 KBWI 091300 RRM
METAR KBWI 091254Z 03007KT 3SM -RA BR BKN009 BKN013 OVC027 03/02 A3013
RMK AO2 SLP203 P0004 T00280017
SAUS41 KBUF 091356
MTRBUF

The forecast editor is implemented as a dialog. It is instantiated from the main GUI of avnwatch.py. The type of forecast (TAF or TWB) has to be selected when the editor is started.

The GUI contains four main areas: menu bar, command buttons, header labels, forecaster selection menu, text editor window and, optionally, METARs window.

Menu Bar

- **File**

Print

Invokes print dialog. The dialog displays a print command: `lp` for HP−UX, `lpr` or `lp` for Linux systems. If you want to redirect output to a particular printer, enter appropriate options (for example: `lp -dlp2`). You can use this dialog to add content of the text window to *fxatext* database: enter textdb −w CCCNNNXXX with the appropriate AFOS header.

Clear

Clears error tags set by the formatting (quality check) action. This selection can be used to force transmission of forecasts that did not pass QC.

Update

Updates issue and valid times in all forecasts within displayed bulletins.

Save

Allows users to save edited forecast to a file. Invokes file selection dialog.

Restore

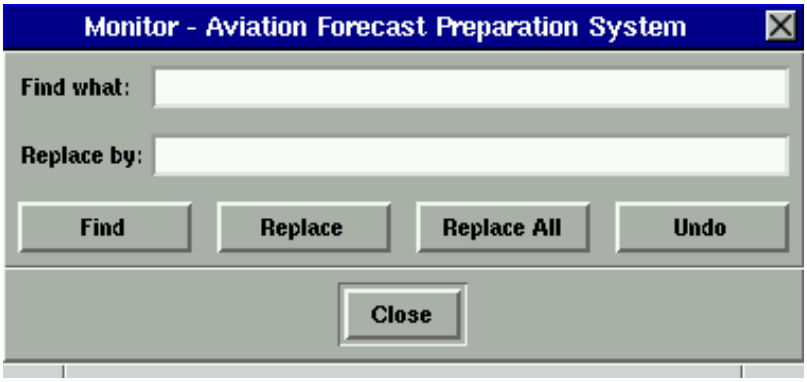
Allows users to restore forecast from a file. Invokes file selection dialog. *Note:* The backup file does not contain information about forecast headers. You must assure that the forecast restored from the backup file matches product name displayed in the button just left to the header time entry above the text window. Press the button to display product selection window, select the product and press **OK**. Otherwise transmission will fail with the error message: *id* does not belong in this product.

- **Edit**

Cut Copy Paste

These three options provide standard editing capabilities.

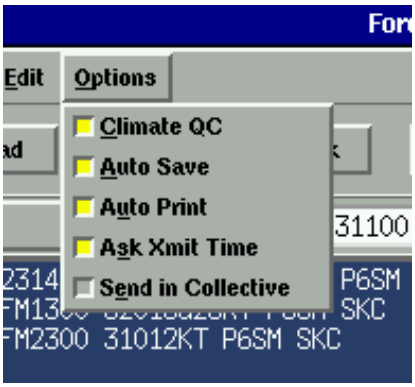
Find



This option invokes the find/replace dialog with undo capabilities.

The **Cut Copy Paste** menu can also be invoked by pressing right mouse button within the text area.

- **Options**



Climate QC

If selected, after syntax check, the TAF weather is compared to the past events in climate database. Unlikely weather combinations are flagged.

Auto Save

When selected, the content of the window is saved every 60 seconds to a backup file in the directory /awips/adapt/avnfps/tmp/. The file is named bulletin.kind where kind is either taf or twb (for example local.taf). When a bulletin is reloaded, the file is renamed to bulletin.kind.prev.

Auto Print

When selected, the forecast will be printed on the default printer when the **Send** button is pressed.

Show Headers

When selected, the displayed METARs will be shown with the WMO headers.

Ask Xmit Time

When selected, a transmission time selection window will pop up when sending regular isse forecasts.

Send in Collective

Normally, the bulletin is split before transmission, so that each TAF is sent separately. When this option is selected, the whole bulletin is transmitted as one file. It is intended for some OCONUS sites.

- **Help**

Key bindings

Displays key bindings valid in the text editor window. These are standard bindings as implemented in Tk. It is possible that some of these are overridden by the window manager.

Usage

Displays help for this dialog.

Command Buttons and Forecaster Menu

Load

Invokes the forecast selection dialog. The bulletin (or product) is selected from the **Bulletins** menu. How the forecasts will be loaded depends on the Load Order selection menu in the dialog. In any case, the forcast residing in the /awips/adapt/avnfps/xmit/pending/. directory (i.e. waiting for transmission) is tried first. If exists, it is then deleted from the transmission queue.

QC/Format

Performs quality check (for TAFs only) and assures proper indentation and maximum line length. If QC fails the forecast, the problem areas will be highlighted. The color corresponds to the severity of the problem. Red means the forecast could not be parsed sucessfully. Orange means there is an error according to Chapter D−31. Green indicates a warning.
If the **A/U** (time auto update) toggle is selected, the issue and valid times for each forecast in the bulletin are updated.

Work

Stores current TAF bulletin to fxatext database. The headers used are set in TAF setup GUI and TWB setup GUI.

Send

Splits the bulletin into separate files, one per site, which are written to the directory /awips/adapt/avnfps/xmit/pending/. The transmission program running on the data server is responsible for actual transmission.
The program will check whether a regular forecast is sent before the transmission window closes. If not, an error dialog is displayed and the forecasts will need to be sent out again as delayed products.

Text

Invokes TAF/TWB/AvnMOS text display for sites associated with the bulletin.

Graph

Invokes TAF/METAR/AvnMOS time series graphical display.

Choose name

While forecasts are sent, the editor includes forecaster id as part of the transmission file. This option menu is used to set the number.

Labels and Toggle Buttons

Below the buttons you will find the following:

TAF
: This label displays whether the editor is configured for TAFs or TWBs.

local
: Displays currently loaded bulletin name

101700
: Displays time as would appear in WMO header

CCX
: Displays *BBB* part od WMO header. The letter *X* indicates that the real version number will be set by the editor at transmission time based on the record of previously transmitted forecasts. Anything else will override the program logic.

Insert
: Insert/Overstrike mode. The default is set in X resources file.

Wrap
: If set, long lines will be wrapped. This does not affect transmitted forecasts.

A/U
: When selected, the **QC/Format** button will auto update issue and valid times for all forecast in the bulletin. This option should be turned on unless there are special reasons (such as program bug) not to do so.
Note: You can always update the times from the **File/Update** menu.

Text editor window

The editor is a standard *Tk* text window. *Tk* implements its own set of editing methods (called *key bindings*). The bindings below are quoted from *Brent B. Welch, Practical Programming in Tcl and Tk, second edition, pp 385−387).*

- **Any−Key** − Insert normal printing characters.
- **Button−1** − Sets the insert point, clear the selection, set focus.
- **Control−Button−1** − Set the insert point without affecting the selection.
- **B1−Motion** − Sweep out a selection from the insert point.
- **Double−Button−1** − Select the word under the mouse.
- **Tripple−Button−1** − Select the line under the mouse.
- **Shift−Button−1** − Adjust the end of selection closest to the mouse.
- **Shift−B1−Motion** − Continue to adjust the selection.
- **Button−2** − Paste the selection, or set the scrolling anchor.
- **B2−Motion** − Scroll the window.
- **Key−Left or Control−b** − Move the cursor left one character. Clear selection.
- **Shift−Left** − Move the cursor and extend the selection.
- **Control−Left** − Move the cursor by words. Clear the selection.
- **Control−Shift−Left** − Move the cursor by words. Extend the selection.
- **Key−Right or Control−f** − Right bindings are analogous to Left bindings.
- **Alt−b or Alt−f** − Same as Control−Left, Control−Right.
- **Key−Up or Control−p** − Move the cursor up one line. Clear the selection.
- **Control−Up** − Move the cursor by paragraph which are group of lines separated by a blank line.
- **Control−Shift−Up** − Move the cursor by paragraph. Extend selection.
- **Key−Down or Control−n** − All Down bindings are analogous to Up bindings.
- **PageUp, PageDown** − Move the cursor by one screen. Clear the selection.
- **Shift−PageUp, PageDown** − Move the cursor by one screen. Extend the selection.
- **Home or Control−a** − Move the cursor to line start. Clear the selection.
- **Shift−Home** − Move the cursor to line start. Extend the selection.
- **End or Control−e** − Move the cursor to line end. Clear the selection.
- **Shift−End** − Move the cursor to line end. Extend the selection.

- **Control−Home** − Move the cursor to the beginning of text. Clear the selection.
- **Control−End** − Move the cursor to the beginning of text. Extend the selection.
- **Control−slash** − Select everything in the text widget.
- **Control−backslash** − Clear the selection.
- **Delete** − Delete the selection, if any. Otherwise delete the character to the right of the cursor.
- **BackSpace or Control−h** − Delete the selection, if any. Otherwise delete the character to the left of the cursor.
- **Control−d** − Delete character to the right of the cursor.
- **Alt−d** − Delete word to the right of the cursor.
- **Control−k** − Delete from cursor to the end of the line. If you are at the end of the line, delete the newline character.
- **Control−o** − Insert a newline but do not advance the cursor.
- **Alt−Delete** − Delete the word to the left of the cursor.
- **Alt−Backspace**
- **Control−t** − Transpose the characters on either side of the cursor.

METAR listing window

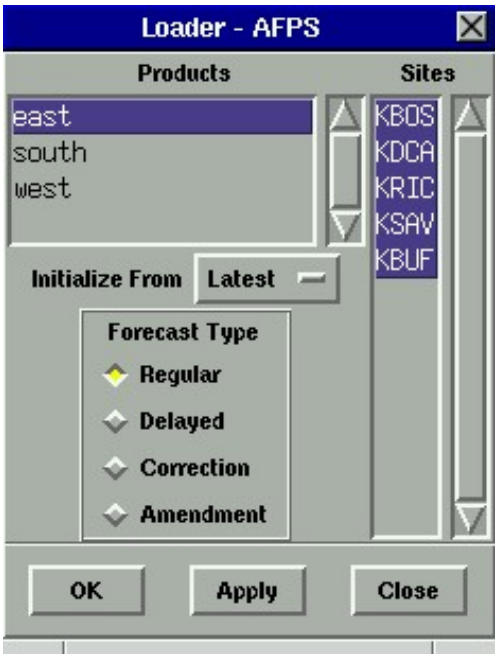
This window is optional. With the default configuration, the window is displayed below the editor window. You can have it to the right of the editor window, or choose not to display it at all. In that case, the popup NumMETARs menu will not be shown. See section How To Modify Resource Configuration File, resource

```
*ForecastEditor*orientation:
```

When a bulletin is loaded, the window will list METARs for all sites contained in the bulletin. A combo box with pull−down arrow can be used to display METARs for a selected site. Number of displayed reports can be changed via the **Num METARs** menu. The **Show Headers** toggle turns on and off display of WMO headers. Depending on the state of **Auto Update**, either the METARs are updated when a new report arrives, or, a **Update** button flags the arrival by changing background color to red. Use the button to refresh display.

How To Load TAF Bulletin

If you press **TAFGen** button in the monitoring GUI, the editor will show up blank. To load one or more bulletins, press the **Load** button.



Highlight one or more products, press the **Select Product(s)** button to get list of sites that are defined in the selected products (bulletins). All sites will be selected when first displayed. You can then select the desired sites.
Use standard mouse/keybord combinations: click on a site selects it. Selection of multiple consecutive sites can be achieved with the mouse press and drag action. To select non−consecutive sites use left mouse button while pressing the <Ctrl> key.
The product list selection is initialize to the products monitored.

Select load order and forecast type. The load order is one of:

Latest
: The previous most recent forecast will be tried first. If it is available (should be under normal circumstances), it will be loaded. The issue and valid time will be initialized to proper values, depending on forecast type. If the **Use METARs** option was selected in the main GUI, the forecast will be updated with the current weather. More precisely, if the most recent METAR is not older than 1 hour, its data will be merged with the forecast.
If the previous TAF is not available, the template file will be loaded instead. Probably this is not what you want.

Previous
: Same behavior as above, but you can select a previously sent forecast: for each site a popup dialog will display a list of recent TAFs. This option is intended to allow to amend TAFs after the next forecast period regular issue has been sent. Hopefully this won't be required.

Template
: The template file will be loaded

Merge
: The previous forecast will be loaded as described above. In addition, the part of the template after valid time (if any) will be appended. This option is only useful for sites with scheduled part−time observations.

Forecast type is one of **Regular**, **Delayed**, **Correction** or **Amended**. Depending on the selected type, the header time and *bbb* entries and the issue and valid time of the forecast(s) loaded will be initialized to appropriate values. You can always change the times and bbb field (i.e. forecast type) later. Use **File/Update** menu button to update issue and valid times within the body of the forecast.

Press **OK** to load the selection. If you have chosen to load backup file, a file selection dialog will be displayed with the

current directory set to /awips/adapt/avnfps/tmp/. In that case, choose **Open**.

If you press **Amd** or **Cor** button in the monitoring GUI, the editor will be shown with most recent forecast initialized from the previous forecast, i.e. the result will be the same as using **Previous** as the load order.

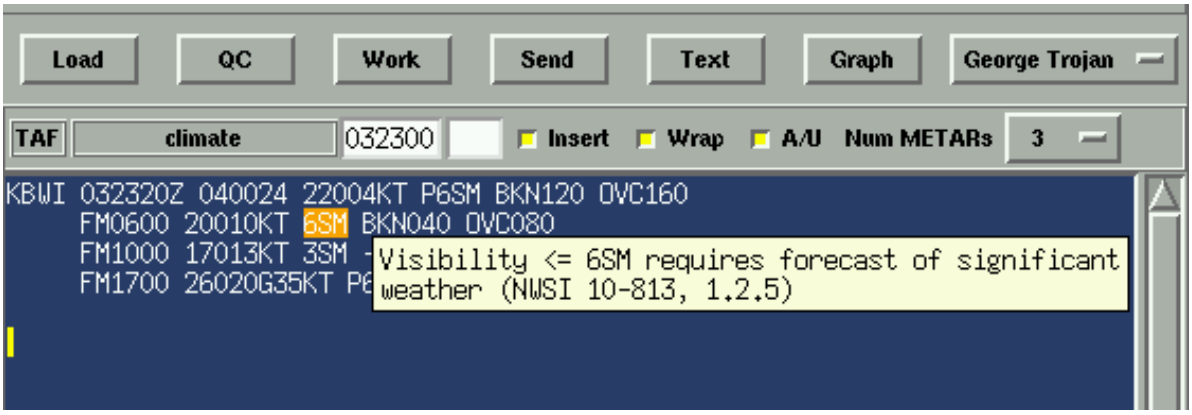
When you load a non−regular issue forecast, the third letter of the **bbb** field is initialized to **X**. The correct version (or versions, if you are editing a collective bulletin) will be obtained from the transmission log file when you press the **Send** button.
In case the previous forecasts were sent from a different system you may have to overwrite the version number. Anything else than **X** indicates the *true* version that will form part of the WMO header. Note that the *true* version will be applied to *all* forecasts within the edited bulletin.

TAF Quality Control

When you are done with forecasts, a quality control check has to be made before forecasts can be sent. Press the **QC/Format** button. The program will attempt to decode all forecasts and reassemble them, assuring proper indentation and maximum line length. Any errors/warnings will be highlighted in the displayed bulletin. There are 3 levels of errors:

- Red − if the decoder cannot determine the meaning of particular word. This is a fatal error, the rest of the forecast (for a given site) is not decoded.
- Orange − this indicates an error as specified by Chapter D−31 (shall not ...).
- Green − this indicates a warning as specified by Chapter D−31 (should not ...).

If you press the left mouse button while pointing at the offending text, an explanatory message will be shown.



The wording of the displayed messages can be changed, if neccessary, by editing the file /awips/adapt/avnfps/etc/catalog.txt.

To display relevant observations and forecasts while typing your own, press the **Display** button. A text display dialog will be shown.

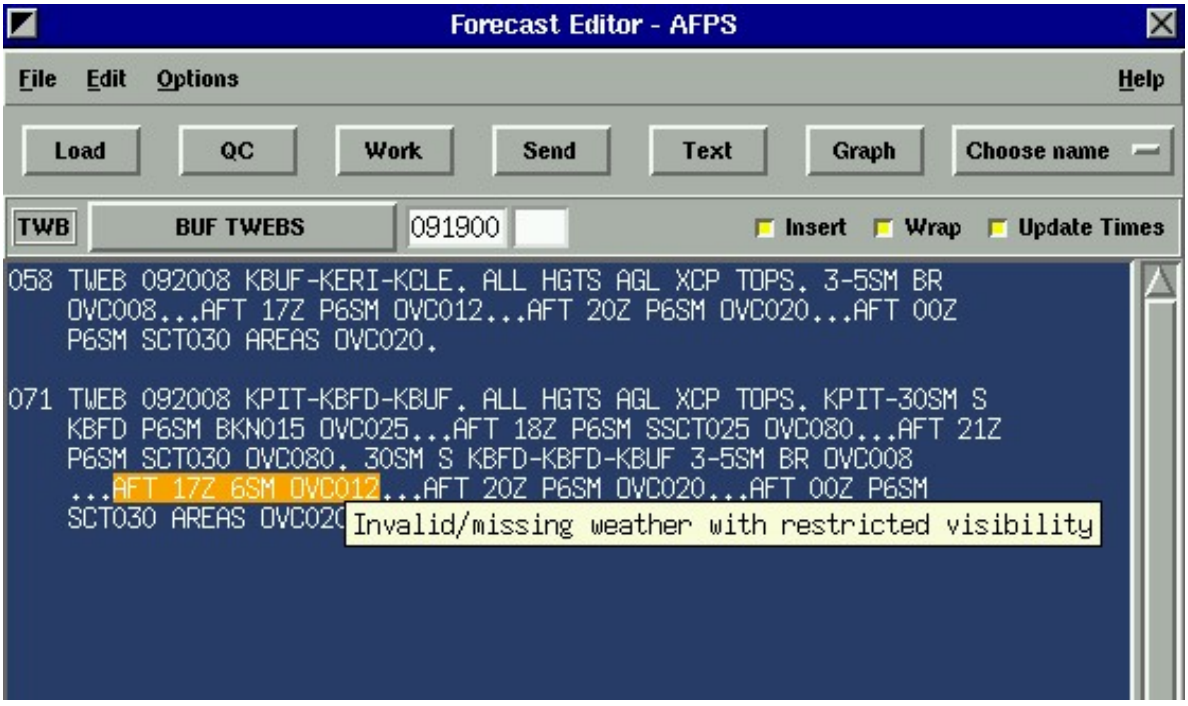
A bulletin that passes quality control without fatal errors is split into separate files, one for each TAF, written to the directory /awips/adapt/avnfps/tmp/*workstation*. These TAFs can be then plotted in the time series display window.

How To Load TWB Bulletin

The TWBs are loaded the same way as TAFs. The only difference is that the **Delayed** forecast type is disabled (i.e. "grayed out"). The reason is to adhere to section 5.2 of Operations Manual, Chapter D−30.

TWB Quality Control

The quality control for TWBs is rudimentary. The forecast is split into phrases, separated by a ".". The first phrase is checked for valid route, keyword "TWEB" and proper time format. The second phrase should contain either "NIL TWEB" or "ALL HGTS ...". The subsequent phrases are checked for weather sequence and time. An attempt is made to check for valid visibility−obstruction to vision combination and for valid sequence of cloud layers. In case of an invalid entry, the whole relevant phrase is highlighted, and the explanatory error message will be displayed the same way as with the TAF QC. There are 2 levels of errors, flagged in orange or green. No checks for valid abbreviation is performed.
Proper formatting, such as maximum line length and indentation, is implemented.



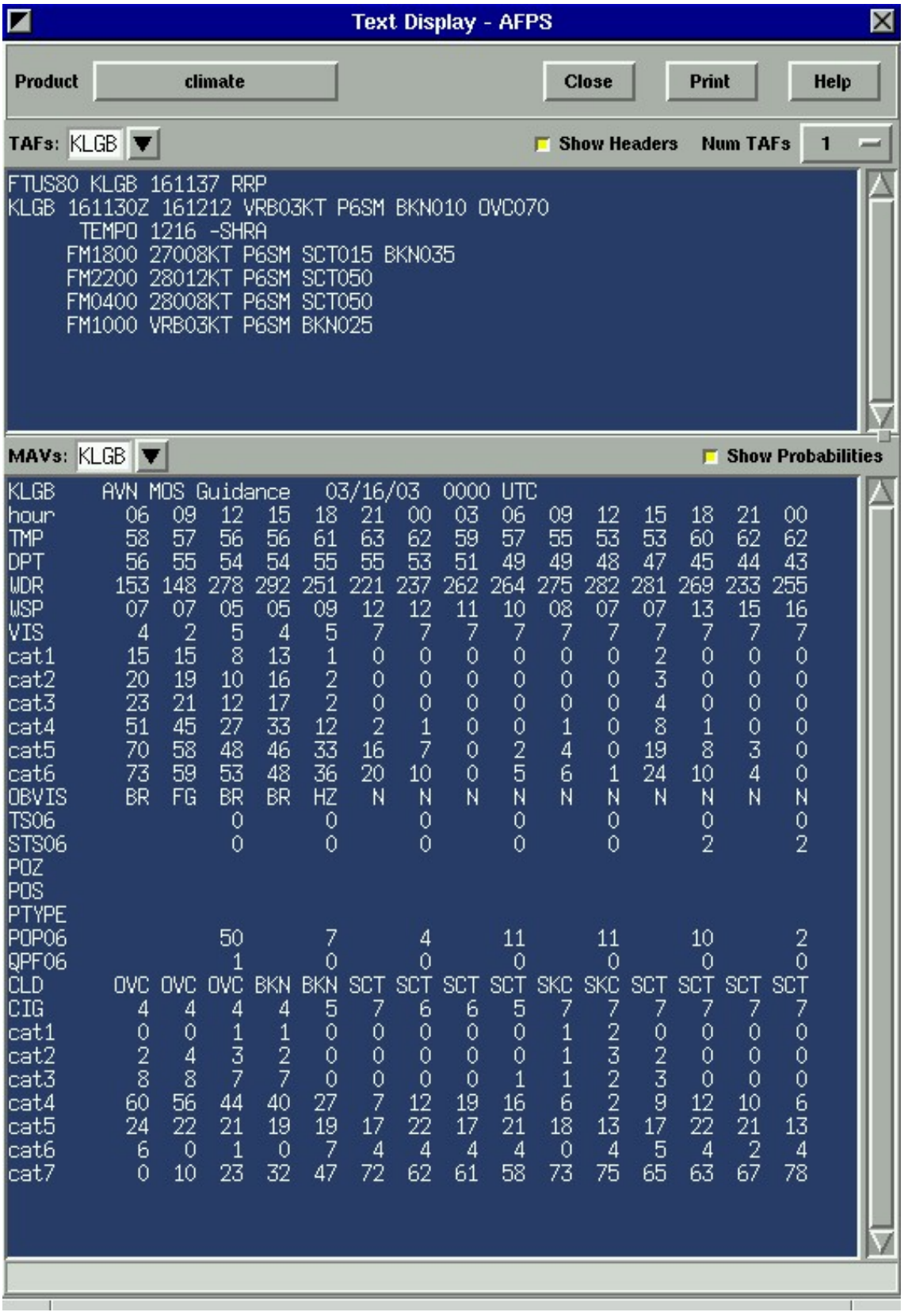
How To Send Forecasts

To send error free forecast, press the **Send** button. The bulletin will be split into files, one per each site that are written to the directory /awips/adapt/avnfps/xmit/pending/ and will be processed by the transmission server.
Amendments, corrections and delayed forecasts are sent immediately. When you press **Send** and the *bbb* field is blank (i.e. you issue a regular forecast), a popup dialog allows you to specify the transmission time. The time is initialized to the beginning of the appropriate transmission window. There is no check whether the time you specify is within that window.

If the forecast contains errors that are above the office mandated threshold specified in the X resources file, you need to clear the errors first. Select **Clear** from **File** menu.
If the forecast is loaded as regular, and the transmission window expires before you were able to send it, a confirmation dialog will appear asking whether you want to send a delayed forecast. This works before the start of valid hour. Otherwise, you need to type RRX in the *bbb* entry field and update times via the **File/Update** menu.

Header times follow WMO rules. That is, regular issue forecast have the header time one hour less start of the valid time. All subsequent corrections and amendments will preserve this value.

Text Display Window



The text dialog displays recent forecast and observations relevant to the edited bulletin(s). The content varies depending on whether TAFs or TWBs are prepared.
The combo box **TAFs** allows for selection of TAFs: *ALL* displays all forecasts in the bulletin, select site id to view TAF

for that site.
The **Show Headers>** button toggles display of WMO headers
The number of METARs can be selected via the **Num METARs** menu.

When called from TAF editor, the bottom part displays AVN MOS guidance. The bulletin is built from data stored in NetCDF files in the directory /data/fxa/point/mos/AVN/netcdf. *Note:* for the pre−OB2 baseline release of AWIPS only 00Z and 12Z forecasts are available.

When called from the TWB editor, the bottom part displays recent forecasts. The **TWBs**, **Show Headers** and **Num TWBs** selections work the same way as in the top part.

To load another bulletin, press the bulletin name pushbutton right to the **TAF/TWB** menu (the one with the *Product selection* balloon message). A bulletin selection dialog will appear.

Select one or more buletins and press **OK**. The bulletin name will be displayed in the pushbutton. When more than one bulletin is selected, the first name will be shown, proceeded by an ellipsis (...).

When you switch between TAFs and TWBs, the bulletin name will be blanked out.

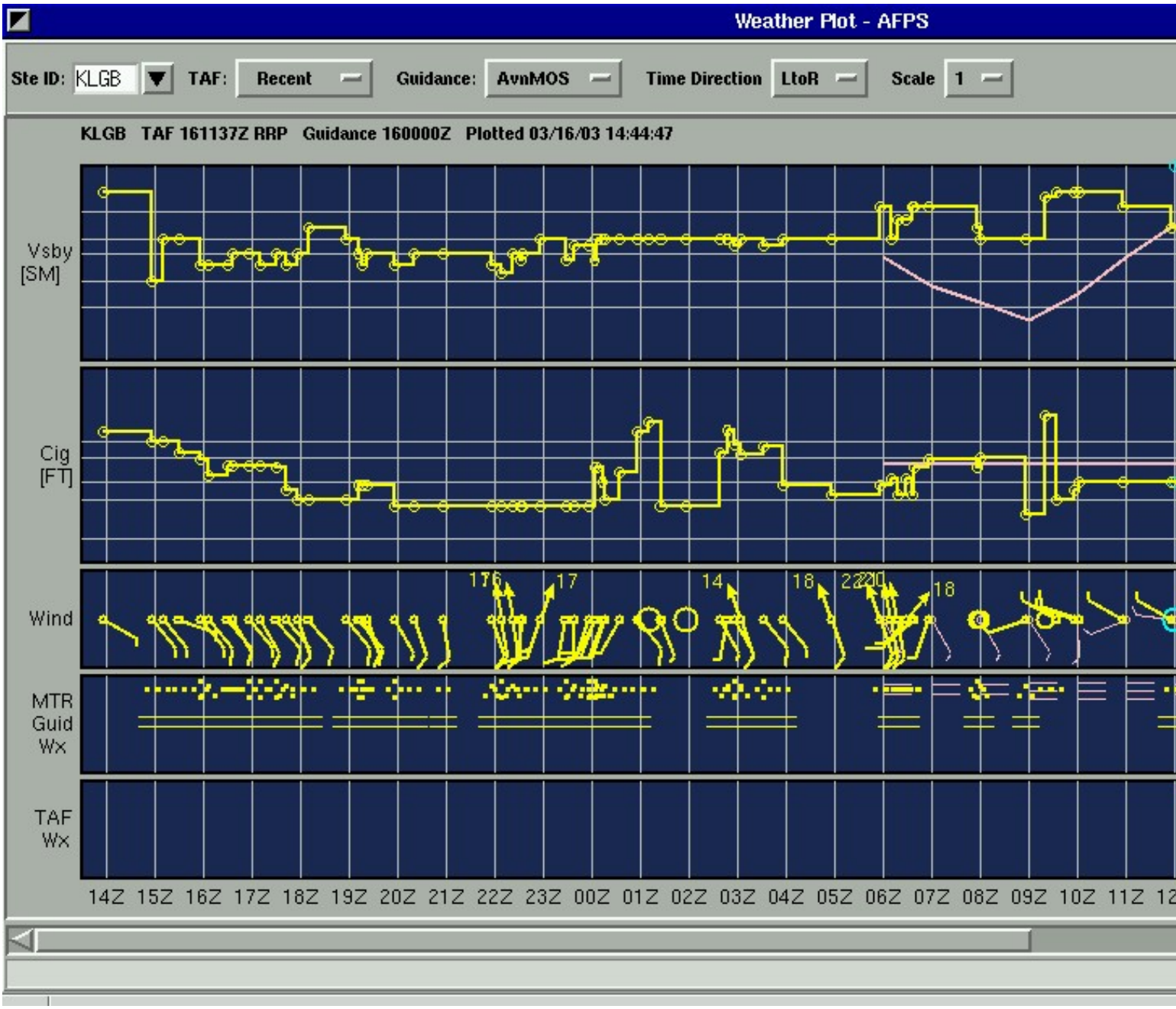
There are three command buttons in this dialog:

Close
: Closes this dialog

Print
: Invokes the print dialog. The dialog displays print command: lp for HP−UX, lpr or lp for Linux systems. If you want to redirect output to a particular printer, enter appropriate options (for example: lp −dlp2). You can use this dialog to dump content of the text window to a file: enter cat > filename to write to filename.

Help
: Displays the help dialog

Time Series Display Window



When invoked, the dialog displays the most recent TAF and METAR data and the Aviation MOS for the selected site in the form of a time series. The time period is: [HH − 24, HH + 24] where HH is the current time rounded down to the current hour. The graph contains 4 areas, ceiling and visibility are plotted as stepwise continuous lines with the small circles indicating the value at the specific time. The TEMPO and PROB groups are plotted with the same color. The BECMG group is treated as a consecutive TEMPO and FM groups. Example:

```
BECMG 1517 OVC030
```

is plotted the same way as would be

```
 TEMPO 1517 OVC030
FM1700 ... OVC030
```

In order to avoid clutter, weather symbols are plotted in two separate windows.

As with the text dialog, once displayed, the forecaster can plot data for a different site by entering its id in the **Site ID** combo box. Use the arrow button to select site for viewing.

A menu for TAF selection contains 3 entries:

- **Recent** − plots the most recent TAF
- **Previous** − plots the previous TAF
- **Edited** − plots the TAF written by forecast editor. Every time the editor runs QC, for each TAF currently in the editor window, a file is written in the directory /awips/adapt/avnfps/tmp/*workstation/* If the file is not older than 6 hours, the TAF is plotted.

The **Guidance** menu contains currently only one option: **AvnMOS** which displays AVN MOS data (from NetCDF files stored in /data/fxa/point/mos/AVN/netcdf). Since the MOS forecast data is every 3 hours (6 for probability data), the values are interpolated. Thunder and precipitation events are determined if the respective probability exceeds certain threshold. Currently the thresholds are coded in the file AvnMosTaf.py and are set to 40% for thunder and 50% for precipitation.

The time increases from left to right. For the benefit of oldtimers, there is an option menu that allows to have the orientation reversed.

The **Scale** menu is used to change horizontal (time) zoom factor. This feature may be useful to reduce clutter of weather symbols when conditions change rapidly, such as in she case shown above.

There are three command buttons in this dialog

Plot
: Reloads data and redraws the display for the current selection.

Print
: Invokes the print dialog. The dialog displays print command: lp for HP−UX, lpr or lp for Linux systems. If you want to redirect output to a particular printer, enter appropriate options (for example: lp −dlp2). You can use this dialog to dump content of the text window to a file: enter cat > filename to write to filename. All the above is exactly the same as in the text display dialog. However the data passed to a printer or stored in a file is first converted to a *Postscript* format. *NOTE:* The output is scaled to fit on a letter size paper. This means that increasing the horizontal zoom factor makes the printed output smaller (that is, with respect to height and font size).

Help
: Displays the help dialog

TAF Decoder Details

The decoder works as a finite state machine. First, it tries to recognize all the tokens (words such as P6SM, or phrases, such as NIL AMD). The order of tokens depends on the TAF group, see section Order of fields. Regular expressions are used for the match, see Required format. If one of the tokens is not recognized, the decoder stops and marks the token as a FATAL error. Each token is then decoded (trivial for site id, valid time has to represent valid hours). Types of checks for individual tokens is described in section Checks done by decoders for individual fields. Finally, there are checks for dependencies within each TAF group (i.e. obscuration vs visibility), see Consistency checks within a group, between groups, Consistency checks between groups and Time periods consistency checks. If a condition is not fulfilled, the failed token is tagged with error (**EMnn**) or warning (**WMnn**). The messages associated with these tags are listed in Message

catalog.

In this section brackets "[]" denote optional fields. Vertical lines mean "or" statements.

Order of Groups

```
MAIN (FM|BECMG|TEMPO|PROB)...
```

Order of Fields

In the following group the fields are defined as follows: id = site identifier itime = issuance time vtime = valid time wx_p = represents precipitation wx_np = represents non−precipitation wx_vc = represents weather in the vicinity. augm = augmentation phrases

```
MAIN:
id  itime vtime (NIL | (wind vis [wx_p] [wx_np...] [wx_vc] sky [shear] [augm]))
FM:
vtime (NIL | (wind vis [wx_p] [wx_np...] [wx_vc] sky [shear]
[augm]))
BECMG TEMPO PROBnn:
vtime [wind] [vis] [wx_p] [wx_np...] [wx_vc] [sky] [shear] [augm]
```

Required Format

In this in section regular expression syntax is used. Multiple lines mean that either one of the expressions must be matched.

```
id:
     [A−Z]{4}
itime:
     [0−3][0−9][0−2][0−9][0−5][0−9]Z
```

where "[0−3]" denotes the first digit of the month can be any digit between 0 and 3 and "[0−9]" denotes that the second digit of the month can be any number between 0 and 9, etc.

```
vtime:
     [0−3][0−9][0−2][0−9][0−2][0−9]
     FM[0−2][0−9][0−5][0−9]
     BECMG [0−2][0−9][0−2][0−9]
     TEMPO [0−2][0−9][0−2][0−9]
     PROB[34]0 [0−2][0−9][0−2][0−9]
wind:
     [0−9][0−9]0[0−9][0−9][0−9]?(G[0−9][0−9][0−9]?)?KT
     VRB[0−9][0−9][0−9]?(G[0−9][0−9][0−9]?)?KT
vis:
     ([0−9][0−9]*|([1−4]|[1−4])?[1357]/([248]|16))(SM)
     P6SM
wx_p:
     ([−+])?(SH|TS|FZ)?(DZ|RA|SN|SG|IC|PL|GR|GS){1,3}
     TS
wx_np:
     ([−+])?(MI|PR|BC|DR|BL|FZ)?
        (SN|BR|FG|FU|VA|DU|SA|HZ|PY|PO|SQ|FC|SS|DS)
```

```
     NSW
wx_vc:
     VC(FG|SH|TS)
sky:
     SKC
     (FEW|SCT|BKN|OVC)[0-9][0-9][0-9](CB)?
     VV[0-9][0-9][0-9](CB)?
shear:
     WS[0-9][0-9][0-9]/[0-9][0-9]0[0-9][0-9][0-9]?KT
augm:
     NIL( AMD)?
     AMD NOT SKED AMDTM_
     AMD LTD TO CLD VIS AND WIND (AMDTM_)?
```

where

```
AMDTM_ = ((AFT[0-2][0-9]Z)|(TIL[0-2][0-9]Z)|([0-2][0-9]Z-[0-2][0-9]Z))
```

The decoder will fail to decode a TAF if the above requirements are not met.

Checks Done by Decoders for Individual Fields

In this section, an expression followed by an arrow means that the expression must be true otherwise an error or warning message will be displayed. The number corresponds to the key value in the file catalogue catalog.txt. For example, in the check for the valid time *vtime* in the following example, if the start hour of valid time is equal to *24* the displayed error message will read Invalid start hour: 24

```
issue time - function itime():
     none
valid time - function vtime():
     [0-3][0-9][0-2][0-9][0-2][0-9]: DDHHhh
     hh > 0 and hh <= 24                              -> Error 21
     HH >= 0 and HH < 24                              -> Error 22
     dd is a valid day for current month              -> Error 23
     vtime.from <= itime + 6hr and vtime.to > itime   -> Error 24

     FM[0-2][0-9][0-5][0-9]: HHMM
     none

     BECMG [0-2][0-9][0-2][0-9]: HHhh
     hh > 0 and hh <= 24                              -> Error 21
     HH >= 0 and HH < 24                              -> Error 22
     end time - start time <= 2hr                     -> Error 26

     TEMPO [0-2][0-9][0-2][0-9]: HHhh
     hh > 0 and hh <= 24                              -> Error 21
     HH >= 0 and HH < 24                              -> Error 22
     end time - start time > 4hr                     -> Error 56

     PROB30 [0-2][0-9][0-2][0-9]: HHhh
     hh > 0 and hh <= 24                              -> Error 21
     HH >= 0 and HH < 24                              -> Error 22
     end time - start time > 6hr                      -> Warning 259

wind - function wind():
     [0-9][0-9]0[0-9][0-9][0-9]?(G[0-9][0-9][0-9]?)?KT:
     dddff[Ggg]KT
```

```
      gg must be in HM database                          -> Error 28
      ff must be in HM database                          -> Error 29
      ddd must be in HM database and !(ddd == 0 and ff > 0)
                                                         -> Error 30
      gg > ff                                            -> Error 32

      VRB[0-9][0-9][0-9]?(G[0-9][0-9][0-9]?)?KT:
      VRBff[Ggg]KT
      gg must be in HM database                         -> Error 28
      gg > ff                                           -> Error 32
      ff + 4 < 3 < ff + 30                              -> Warning 261

low level wind shear - function shear():
      WS[0-9][0-9][0-9]/[0-9][0-9]0[0-9][0-9][0-9]?KT:
      WShhh/dddffKT
      if (group == BECMG|TEMPO|PROB)                     -> Error 33
      ff must be in HM database                          -> Error 34
      ddd must be in HM database                         -> Error 35
      hhh must be in HM database                         -> Error 36
visibility - function vis():
      ([1-9][0-9]*|([1-4]|[1-4])?[1357]/([248]|16))(SM):
      vvvvSM
      vvvv must be in HM database                       -> Error 37

      M1/4SM
      none

      P6SM
      none
significant weather, precipitation - function wx_p():
      ([-+])?(SH|TS|FZ)?(DZ|RA|SN|SG|IC|PL|GR|GS){1,3}
      The [descriptor]phenomenum string must be in HM
          database                                      -> Error 38

significant weather, thunder - function wx_ts():
      TS
      none

significant weather, non-precipitation phenomena - function wx_np():
      ([-+])?(MI|PR|BC|DR|BL|FZ)?
             (SN|BR|FG|FU|VA|DU|SA|HZ|PY|PO|SQ|FC|SS|DS)
      The [descriptor]phenomenum string must be in HM
           database                                     -> Error 38
      FC should not be forecast                         -> Warning 260
      SN must have a descriptor                         -> Error 45

no significan weather - function wx_nsw():
      NSW
      if (group == MAIN|FM|PROB)                        -> Error 39

weather in the vicinity - function wx_vc():
      VC(FG|SH|TS)
      none

clear sky - function sky_clr():
      SKC
      none

sky condition - function sky():
      (FEW|SCT|BKN|OVC)[0-9][0-9][0-9](CB)?: NNNhhh[CB]
      NNN must be in HM database                        -> Error 40
      hhh must be in HM database                        -> Error 41
```

```
      hhh == 0 and NNN != OVC                          -> Error 42

      VV[0-9][0-9][0-9](CB)?: VVhhh[CB]
      hhh must be in HM database                      -> Error 41

   vertical visibility - function sky_vv(): VVhhh[CB]
      hhh must be in HM database                      -> Error 41

   augmented data - function augm():
      NIL( AMD)?
      none

      AMD NOT SKED
      AMD NOT SKED AMDTM_
      AMD LTD TO CLD VIS AND WIND AMDTM_
      if AMDTM_: AFThhZ
      hh < 24                                         -> Error 21

      if AMDTM_: TILhhZ
      hh < 24                                         -> Error 21

      if AMDTM_: HHZ-hhZ
      HH < 24                                         -> Error 21
      hh > 0 and hh <= 24                              -> Error 22
```

Time Periods Consistency Checks

```
   FM group
      vtime.from > main.vtime.from                    -> Error 49

   BECMG TEMPO PROB
      vtime.from >= main.vtime.from and vtime.to
                    <= main.vtime.to                  -> Error 25

   FM BECMG
      prev_group == MAIN
           none
      prev_group == FM
           vtime.from > prev.vtime.from               -> Error 44
      prev_group == BECMG
           vtime.from > prev.vtime.to                 -> Error 51
      prev_group == TEMPO|PROB
           vtime.from >= prev.vtime.from              -> Error 50

   TEMPO
      prev_group == MAIN
           none
      prev_group == FM
           vtime.from >= prev.vtime.from              -> Error 44
      prev_group == BECMG|TEMPO|PROB
           vtime.from >= prev.vtime.to                -> Error 52

   PROB
      prev_group == MAIN
           none
      prev_group == FM
           vtime.from >= prev.vtime.from              -> Error 44
      prev_group == BECMG|TEMPO|PROB
           vtime.from >= prev.vtime.to                -> Error 52
```

Consistency Checks Within a Group

```
wx
     NSW can be the only weather element in a group  -> Error 47
     TEMPO
          must not contain weather in the vicinity   -> Error 43
     VC descriptor should be the last entry in
          the wx group                               -> Warning 257
     TS requires wind forecast                       -> Error 16
     TS requires CB in the sky groups                -> Error 11
     PROB
          must contain precipitation or TS           -> Error 16

wind:
     VRB wind > 6KT requires convective
                           weather (TS, SH)          -> Error 31

sky
     number of sky fields must be < 3               -> Warning 258
     for any consecutive cloud layers
          sky1.amt == SKC || sky2.amt == SKC not allowed
          sky1.base < sky2.base
          sky1.amt != VV and sky1.amt != OVC
          sky1.amt == SCT and sky2.amt == FEW not allowed
          sky1.amt == BKN and sky2.amt == FEW|SCT not allowed

wx and visibility
     vis > 6SM
          group == TEMPO|BECMG require NSW           -> Error 46
          only DRDU, DRSA, DRSN, MIFG, PRFG, BCFG
          can be reported                            -> Error 18
     vis <= 6SM
          weather forecast required (no NSW)         -> Error 12
     +PL => vis < 3SM                                -> Error 53
     if descriptor not DR or BL:
        +SN|+DZ => vis <= 1/4SM
        SN|DZ >= vis <= 1/2SM
        +SS|+DS >= vis < 5/16SM                      -> Error 54
     BR => vis >= 5/8SM and vis <= 6SM               -> Error 15
     FG|FZFG but not VC => vis < 5/8SM
     MIFG >= vis >= 5/8SM                            -> Error 14
     SS|DS => vis <= 5/8SM and vis >= 5/16SM         -> Error 55
```

Consistency Checks Between Groups

```
     BECMG and TEMPO/PROB groups, in any order,
     will not be used consecutively                  -> Error 20
```

AVNFPS File Format

Release 1.0
June 19, 2003

System Files

The system−wide files are located in the directory /awips/adapt/avnfps/etc, with the exception of X resources file, which exists in subdirectory app−resources.

Forecasters

etc/forecasters

An ASCII file, containing list of forecaster names, to appear in the startup menu and associated numbers, used to form transmission file names.
Lines starting with # are comments.
Format:
number name

Example:

```
# etc/forecasters
# List of forecasters
# Format:
# number name
1 George Trojan
2 Mike Graf
3 Dave Hotz
```

Error Catalog

etc/catalog.txt

Each entry consists of 2 or more lines:
First line:

KEY Num Severity
where *KEY* is a keyword, *Num* is message number corresponding to the values hardcoded in TAF decoder (source file lib/src/avn/tafdecoderP.C). The numbers 10 − 255 are errors, 256 − 511 are warnings.
Second and subsequent lines: *Message Message* is the thext that will be displayed in the balloon popup window in the main monitoring GUI. The maximum line length should not exceed 40−50 characters. If possible, a reference to the relevant section of NWSI should be included.

Current content:

```
# file: catalog.txt
# error message catalog for TAF decoder
# keys < 256 are errors, > 256 are warnings
# best left alone: keys are hardcoded in the file tafdecoderP.C
# last update: January 16, 2003
KEY     11      Error
Thunderstorm forecast requires CB in the cloud
group (NWSI 10−813, 1.2.7.3)
KEY     12      Error
Visibility <= 6SM requires forecast of significant
weather (NWSI 10−813, 1.2.5)
KEY     13      Error
Volcanic ash requires visibility forecast
(NWSI 10−813, 1.2.6)
KEY     14      Error
FG or FZFG forecast requires visibility < 5/8SM,
MIFG requires visibility >= 5/8SM (NWSI 10−813, 1.2.6)
KEY     15      Error
BR forecast requires visibility between 5/8SM
and 6SM (NWSI 10−813, 1.2.6)
KEY     16      Error
PROB group must include forecast of a thunderstorm
or precipitation event (NWSI 10−813, 1.2.9.4)
KEY     17      Error
Invalid sky cover sequence (NWSI 10−813, 1.2.7.1)
KEY     18      Error
Invalid weather with visibility >= 6SM
(NWSI 10−813, 1.2.5)
KEY     19      Error
BECMG and TEMPO/PROB groups, in any order, will not
be used consecutively (NWSI 10−813, 1.2.9.1)
KEY     20      Error
Cannot determine datetime
KEY     21      Error
Invalid end hour
KEY     22      Error
Invalid start hour
KEY     23      Error
Invalid day
KEY     24      Error
Issue and valid times do not match
KEY     25      Error
Group time period not within TAF forecast period
KEY     26      Error
Invalid period for BECMG group (NWSI 10−813, 1.2.9.2)
KEY     27      Error
The PROB group shall not be used in the first
9 hours of the valid TAF forecast
(NWSI 10−813, 1.2.9.4)
KEY     28      Error
Invalid value of wind gust
```

```
KEY     29      Error
Invalid value of wind speed
KEY     30      Error
Invalid value of wind direction
KEY     31      Error
Variable wind speed must be between 1 and 6KT without
convective actovity.
(NWSI 10-813, 1.2.4)
KEY     32      Error
Wind gust <= wind speed
KEY     33      Error
Forecast of non-convective low-level wind shear
shall not be included in TEMPO or PROB groups
(NWSI 10-813, 1.2.8)
KEY     34      Error
Invalid value of wind shear magnitude
KEY     35      Error
Invalid value of wind shear direction
KEY     36      Error
Invalid value of wind shear height
KEY     37      Error
Invalid value of visibility (NWSI 10-813, 1.2.5)
KEY     38      Error
Invalid value of sig weather (NWSI 10-813, B-4)
KEY     39      Error
Cannot have NSW in this group (NWSI 10-813, 1.2.6)
KEY     40      Error
Invalid cloud amount (NWSI 10-813, 1.2.7.1)
KEY     41      Error
Invalid cloud base (NWSI 10-813, 1.2.7.1)
KEY     42      Error
Cannot forecast partial obscuration
(NWSI 10-813, 1.2.7.2)
KEY     43      Error
TEMPO groups shall not include forecasts of
significant weather in the vicinity
(NWSI 10-813, 1.2.9.3)
KEY     44      Error
Valid time must be > valid time of previous FM
group
KEY     45      Error
Precipitation elements must be encoded in
the first significant weather group
(NWSI 10-813, 1.2.6)
KEY     46      Error
P6SM NSW needed in this group (NWSI 10-813, 1.2.6)
KEY     47      Error
After NSW no further significant weather types
are allowed in this group (NWSI 10-813, 1.2.6)
KEY     48      Error
When reduction in visibility is forecast to change
in TEMPO group, the significant weather causing
the deterioration shall be included
(NWSI 10-813, 1.2.9.3)
KEY     49      Error
Valid time of FM group must be > valid time for
the entire TAF
KEY     50      Error
Valid time must be >= valid time of previous TEMPO
or PROB group
KEY     51      Error
Valid time must be > valid time of previous BECMG
```

```
group
KEY     52      Error
Valid time must be >= end of valid time of previous
group
KEY     53      Error
+PL requires visibility < 3SM
PL requires visibility <= 6SM (FMH#1, p 8-3)
KEY     54      Error
+SN or +DZ requires visibility <= 1/4SM
SN or DZ requires visibility <= 1/2SM
(FMH#1, p 8-3)
KEY     55      Error
+SS ot +DS requires visibility <= 5/16SM
SS or DS requires visibility <= 5/8SM
(NWSI 10-813, B-4)
KEY     56      Error
The period covered by a TEMPO group will not
exceed 4 hours (NWSI 10-813, 1.2.9.3)
KEY     57      Error
The PROB30 group will not be used as a modifier
of BECMG or TEMPO (NWSI 10-813, 1.2.9.4)
KEY     58      Error
Repeated occurence of weather elements
KEY     257     Warning
Forecast weather in the vicinity should be
the last entry in the weather group
(NWSI 10-813, 1.2.6.1)
KEY     258     Warning
Number of cloud groups should not exceed three
(NWSI 10-813, 1.2.7.1)
KEY     259     Warning
The period of time covered by a PROB should be
6 hours or less (NWSI 10-813, 1.2.9.4)
KEY     260     Warning
Tornadoes or Waterspouts should not be
forecast in terminal forecasts
(NWSI 10-813, B-4)
KEY     261     Warning
Suspicious value of wind gust
```

X Resources

The file etc/app-defaults/X provides system-wide X resources. Format follows standard X syntax, comment lines start with an !. Individual entries should not be commented out, order is important, as it is processed by the Resource Editor dialog to create personal files X.*number*, where *number* is the forecaster number from etc/forecasters.

Current content:

```
!! default X resources configuration file
!! last modified January 14, 2003
!! WARNING: Do NOT change order or delete/comment out entries.
!! user editable options
!! modifies behavior of dialogs
*transientDialogs:              0
!! confirmation on closing editor
*confirmClose:                  1
!! alert options: use colors for the first level that should result in
!! notification - pale green, yellow, orange, red, purple or none
*notifyDeiconify:               yellow
```

```
*notifyRaise:                   yellow
*notifyPlay:                    pale green
!! not implemented
!!*notifyTalk:                    none
*playFile:                      /awips/fxa/data/sounds/asterisk.au
!! warning/error level to disallow send:
!! 'Always', 'Warning, 'Error', 'Fatal'
*disallowSend:                  Error
!! forecast editor options
!! use template, one of: Template, Merge, Previous
*loadOrder:                     Merge
!! update TAF with current METAR while loading previous forecast
*usemetar:                      1
!! periodically save bulletins in a backup file
*autosave:                      1
!! update issue and valid times on QC
*autoupdate:                    1
!! print forecasts on send
*autoprint:                     1
!! do climate QC (Linux only)
*climate:                       1
!! time between consecutive checks in seconds
*timeout:                       30
!! number of TAFs to display
*numTafs:                       1
!! number of METARs to display
*numMetars:                     3
!! show bulletin headers in the text window
*showHeaders:                   1
!! show categoy probapilities for AVN MOS
*showProbs:                     1
!! fonts
!! change 120 to 140 if you want bigger default font
*font:                          -adobe-helvetica-bold-r-normal-*-*-100-*-*-p-*-iso8859-9
!! text window font
*Text.font:                     7x14
!! listbox font
*Listbox.font:                  7x14
!! entry font
*Entry.font:                    7x14
!! forecast editor font - may want bigger
*ForecastEditor*Text.font:      7x14
!! balloon message (popup) font
*balloonFont:                   7x14
!! colors
!! global options
*background:                    grey70
*foreground:                    black
!! text window colors
*Text.background:               #2f3f6f
*Text.foreground:               #ffffff
*insertbackground:              yellow
!! entry window colors
*Entry.background:              white
*MessageBar.Entry.background:   grey85
!! forecast editor colors - may want different
*ForecastEditor*Text.background:        #2f3f6f
*ForecastEditor*Text.foreground:        #ffffff
!! time series plot
*WxPlot*tafForeground:          cyan
*WxPlot*mosForeground:          pink
*WxPlot*metarForeground:        yellow
```

```
*WxPlot*plotBackground:         #142850
!! other text window  specific options
*Text.width:                    72
*Text.height:                   24
*TextDisplay*Text.height:       18
*ForecastEditor*Text.width:     72
!! horizontal, vertical or none
*ForecastEditor*orientation:    vertical
!! cursor width
*insertWidth:                   5
!!
!! editable by knowledgable people only
!!
!! send forecasts in a collective
*collective:                    0
!!
!! used by avnsetup only
!! maximum number of sites in a collective forecast
!! those may need to be increased before attempting to run avnsetup
!! if you have a lot of TAFs
*TAFEntry*numrows:              24
*TWBEntry*numrows:              24
*Trigger*numrows:               42
!! font in the main GUI
*topfont:                       -adobe-helvetica-medium-r-normal-*-*-240-*-*-p-*-iso8859-9
!! prevents line wrap in message boxes
*wrapLength:                    0
!! end of X resources configuration file
```

AVN MOS Aliases

AvnFPS assumes that the site ids in AVN MOS match TAF ids. There are however few exceptions, affecting WFOs in Pacific Region and Puerto Rico. The file etc/AVNMOS.alias overrides the default behavior. The format is:

```
metar-id  avnmos-id
```

where avnmos-id can be set to **none** if MOS data is not available. This will prevent the program from displaying popup error messages about missing data. The current content is:

```
PHJR PHNA
TNCM none
TKPK none
```

Climate QC

The two files used by Climate QC module are etc/climcheck.conf and etc/climcheck.list.

Climate quality control is under development and the files are subject to change. etc/climcheck.conf contains four sections.
The first section determines matching ranges for visibility, ceiling, wind speed and direction. Given forecasted value v, the range of values that match is in the interval

```
[max(min(v/factor, v-delta), 0), max(v*factor, v+delta)]
```

where factor and delta are listed after the keyword, i. e.

```
visibility      factor  delta
```

The second section specifies categories that are used to display probabilities (frequencies) of past events. the format is

```
Keyword  value0 value1 ... valueN
```

The third section contains the percentile used to determine the most likely values of ceiling and visibility. Format:

```
Percentile   prevailing         occasional
```

where *prevailing* and *occasional* are in percent.
The last section consists of 2 items,

```
RareEvent       value
```

denotes threshold (%) under which a rare event is signalled. The

```
Unlikely        value
```

line contains threshold used to flag unlikely combination of weather types. If the ratio P(Wx1 and Wx2) / P(Wx1) * P(Wx2) is less than value, the combination of Wx1 and Wx2 is considered unlikely.

Current content:

```
# climcheck.conf
# last update: 03/16/03
###  matching criteria
# for visibility, ceiling and wind speed the matching range is set to:
# [max(min(v/factor, v-delta), 0), max(v*factor, v+delta)]
# visibility: if v > 6.5SM the upper limit is 100SM
# ceiling: if unlimited, only events with unlimited ceiling are matched
#          if > 12000FT, the upper limit is set to 80000FT
# for wind direction, the range is [v-delta, v+delta] (factor not used)
#               factor  delta
visibility      1.5     0.3
ceiling         1.5     300
wind_speed      2.0     3
wind_dir        1.0     30
#
###  Categories
VisCat          0.0  0.5  1.0  2.0  3.0  6.0  100.0
CigCat          0  200  600  1000  2000  3100  8000  90000
#
###  used to select suggested visibility/ceiling
#               PREV    OCNL
Percentile      60      40
#
### threshold used to display warning for unlikely combination and rare events
Unlikely        0.5     # 1 means independent events
RareEvent       0.03    # (% of total cases)
```

The file etc/climcheck.list provides mapping between station id and the data file(s) that should be used for climate QC. The format is:

```
CCCC    CCC1 ...
```

There can be more than one data file for a station, in case when climate at several sites is similar.

Example content:

```
# file: climcheck.list
#
KDCA    IAD
KIAD    IAD BAL
KBWI    BAL
KBUF    BUF
PAVD    VDZ
```

TAF Configuration

For each product *productname* several configuration files are created by the setup program.

Product Definition

etc/tafs/*productname*.def
This file contains list of sites and associated AFOS and WMO headers. Format:

```
   list of tuples (id, AFOS, WMO)
```

id equal to '' (empty string) defines dummy headers used to store bulletin in the Informix database.

Example:

```
[('', 'WRKTAFEAS', 'TTAA00 KMDL'),
 ('KBOS', 'BOSTAFBOS', 'FTUS41 KBOX'),
 ('KDCA', 'WBCTAFDCA', 'FTUS41 KLWX'),
 ('KRIC', 'WBCTAFRIC', 'FTUS41 KAKQ'),
 ('KSAV', 'ATLTAFSAV', 'FTUS42 KCHS'),
 ('KBUF', 'BUFTAFBUF', 'FTUS41 KBUF')]
```

Site Lists

File etc/tafs/*productname*.taf.list contains list of TAF ids associated with the product.
Format:

```
   tuple (site list, dictionary id -> AFOS id)
```

The site list is used to order entiries in the forecast. AFOS ids are needed for the triggers.

Example:

```
(['KBOS', 'KDCA', 'KRIC', 'KSAV', 'KBUF'],
 {'KDCA': 'WBCTAFDCA',
  'KRIC': 'WBCTAFRIC',
  'KBOS': 'BOSTAFBOS',
  'KBUF': 'BUFTAFBUF',
  'KSAV': 'ATLTAFSAV'})
```

File etc/tafs/*productname*.mtr.list contains list of METAR ids associated with the product.
Format:

```
   tuple (site list, dictionary id -> AFOS id)
```

The site list is not used. *Note:* This may have to be changed if METAR id differs from TAF id.

Example:

```
(['KBOS', 'KDCA', 'KRIC', 'KSAV', 'KBUF'],
 {'KDCA': 'WBCMTRDCA',
  'KRIC': 'WBCMTRRIC',
  'KBOS': 'BOSMTRBOS',
  'KBUF': 'BUFMTRBUF',
  'KSAV': 'ATLMTRSAV'})
```

File etc/tafs/*productname*.mav.list contains list of AVN MOS guidance ids associated with the product.
Format:

```
   tuple (site list, dictionary id -> None)
```

Only the keys of the dictionary are used to extract guidance from NetCDF file.

Example:

```
(['KBOS', 'KDCA', 'KRIC', 'KSAV', 'KBUF'],
 {'KDCA': None,
  'KRIC': None,
  'KBOS': None,
  'KBUF': None,
  'KSAV': None})
```

Templates

The 4 files etc/tafs/*productnameHH*.templ where *HH* is one of: 00, 06, 12, 18 are created during setup and can be edited afterwards. At a minimum, each entry contains site id, valid and issue timestapms. The day part is initialized while loading the forecast from template.
Format:

```
   CCCC DDHH00Z DDHHhh text =
```

The text part may continue through several lines. The entries are separated either by the = sign or by a blank line.

Monitoring Rules

The file etc/tafs/XXXX.rules defines default set of monitoring rules. This is a pickled Python dictionary that can be modified by the SiteInfoDialog. Depending on the value of **_Use_Binary** variable, defined in py/Avn.py, the format is binary or ASCII. Each site with id *CCCC* can have its own specialized set of rules stored in a file etc/tafs/*CCCC*.rules.
Format:

```
Class SiteData:
        id:             string
```

```
        runway:         list (length 2) of integers
        vis:            list of floats
        cig:            list of integers
        ruleset:        list of dictionaries (rules)
Rule:  dictionary
        active:         0 or 1
        args:           dictionary (Args)
        color:          string
        unique:         0 or 1
        type:           wx, vsby, wnd or cig
        method:         function, defined in TAFCheck.py
Args:  dictionary
        msg             string
        other key       rule dependent values
        ... (up to 4 keys)
```

TWEB Configuration

For each product *productname* several configuration files are created by the setup program.

Product Definition

etc/twbs/*productname*.def
This file contains list of sites and associated AFOS and WMO headers. Format:

```
   list of tuples (id, AFOS, WMO)
```

Example:

```
[('058', 'BUFTWB058', 'FRUS41 KBUF'),
 ('071', 'PITTWB071', 'FRUS41 KBUF')]
```

Site Lists

File etc/twbs/*productname*.twb.list contains list of TWEB ids associated with the product.
Format:

```
   tuple (site list, dictionary id -> AFOS id)
```

The site list is used to order entiries in the forecast. AFOS ids are needed for the triggers.

Example:

```
(['058', '071'],
 {'058': 'BUFTWB058',
  '071': 'PITTWB071'})
```

File etc/twbs/*productname*.mtr.list contains list of METAR ids associated with the product.
Format:

```
   tuple (site list, dictionary id -> AFOS id)
```

The site list is not used.

Example:

```
(['KBUF', 'KDKK', 'KERI', 'KHZY'],
 {'KDUJ': 'PITMTRDUJ',
  'KDKK': 'BUFMTRDKK',
  'KHZY': 'CLEMTRHZY',
  'KBUF': 'BUFMTRBUF'})
```

File etc/twbs/*productname*.taf.list contains list of TAF ids associated with the product.
Format:

```
   tuple (site list, dictionary id -> None)
```

Only the keys of the dictionary are used to extract guidance from NetCDF file.

Example:

```
(['KBUF', 'KERI', 'KPIT', 'KBFD'],
  'KPIT': 'PITTAFPIT',
  'KBFD': 'PHLTAFBFD',
  'KBUF': 'BUFTAFBUF',
  'KERI': 'CLETAFERI'})
```

Templates

The file etc/tafs/*productname*.templ is created during setup and can be edited afterwards. At a minimum, each entry contains route number, keyword *TWEB* and the timestamp.
Format:

```
   NNN TWEB DDHHhh text
```

The day part of the timestamp is initialized while loading forecast from the template. The text part may continue through several lines. Each entry is separated by a blank line.

Trigger Configuration

Templates

The template, etc/triggerTemplate is created by the setup program. The format is such that the localization script, /awips/fxa/data/localization/scripts/mainScript.csh can proccess it. This file must be located in the directory /awips/adapt/avnfps/etc.
Format:

```
   CCCNNNXXX | | | trigger |
```

where *CCCNNNXXX* is the AFOS id (and file name created by AWIPS decoder) and *trigger* is the executable used to copy the file to the data directory.

Current content (few lines):

```
SEAMTRSEA | | | /awips/adapt/avnfps/bin/HP-UX/mtrtrigger |
CLEMTRCLE | | | /awips/adapt/avnfps/bin/HP-UX/mtrtrigger |
WBCTAFDCA | | | /awips/adapt/avnfps/bin/HP-UX/taftrigger |
SFOTAFPRB | | | /awips/adapt/avnfps/bin/HP-UX/taftrigger |
BOSTWB005 | | | /awips/adapt/avnfps/bin/HP-UX/twbtrigger |
WBCTWB032 | | | /awips/adapt/avnfps/bin/HP-UX/twbtrigger |
```

Storage

An optional configuration file, etc/trigger.conf can be used to change the number of hours of forecasts and observations stored in the data directory.

Current content:

```
# trigger.conf
# this optional file defines number of hours of reports to keep
# default is 25
30
```

AVNFPS FAQs/Troubleshooting

Release 1.0
June 19, 2003

Transmission Problems (How do I determine whether or not the server is running?)

To determine whether the server is running:

(i) Check the color of "Last Pending Check" button in the Monitor GUI and date/time displayed in this button?

green: OK
orange: last xmit/pending directory check was done between 1 and 2 minutes. Might indicate that the server just stopped, or is hung.
red: last check was done more than 2 minutes ago. Most likely a problem.

(ii) On ds, execute:

```
ds1−nmtw{awipsusr}2: ps −ef | grep avn
awipsusr 25032     1  0 15:24:30 ?        0:00 ./avnpython
/awips/adapt/avnfps/1.0/py/avnxmitserv.py
```

The line above indicates that the server is running.

```
ds1−nmtw{awipsusr}3: cd /awips/adapt/avnfps/1.0/xmit
ds1−nmtw{awipsusr}4: ll
total 0
drwxrwxr−x   2 fxa       fxalpha        96 May 15 18:09 sent/
drwxrwxr−x   2 fxa       fxalpha        96 May 15 18:09 bad/
drwxrwxr−x   2 fxa       fxalpha        96 May 21 14:40 pending/
−rwxrwxr−x   1 awipsusr   fxalpha         0 May 25 15:26 tstamp*
```

To start the server, press "Last Pending Check" button in the Monitor GUI.

If the server is running, but the products are not sent. [Return to Top]

(i) Check server log files:

```
ds1−pbz{awipsusr}1: cd /awips/adapt/avnfps/RAP/logs
ds1−pbz{awipsusr}2: ll
total 182
−rw−rw−r−−   1 textdemo   fxalpha      10785 May 20 23:38 XmitServ_Tue
−rw−rw−r−−   1 textdemo   fxalpha      13599 May 21 23:41 XmitServ_Wed
−rw−rw−r−−   1 textdemo   fxalpha       9318 May 22 23:28 XmitServ_Thu
−rw−rw−r−−   1 textdemo   fxalpha       9867 May 23 23:39 XmitServ_Fri
−rw−rw−r−−   1 textdemo   fxalpha      13936 May 24 23:41 XmitServ_Sat
−rw−rw−r−−   1 textdemo   fxalpha       8403 May 25 23:24 XmitServ_Sun
−rw−rw−r−−   1 textdemo   fxalpha       9852 May 26 17:31 XmitServ_Mon
ds1−pbz{awipsusr}3: more XmitServ_Mon
. . . .
SUCCESS: 16:39:48 014−KPBZTAFMGW−FTUS41−KPBZ−0305261100−AAA−1053965415
SUCCESS: 17:30:31 014−KPBZTAFPIT−FTUS41−KPBZ−0305261700−___−1053970200
SUCCESS: 17:30:38 014−KPBZTAFAGC−FTUS41−KPBZ−0305261700−___−1053970200
. . . .
```

The word SUCCESS indicates that the server successfully invoked handleOUP.pl.

(ii) Check corresponding entry in handleOUP.pl log file: [Return to Top]

```
ds1−pbz{awipsusr}4: logs
ds1−pbz{awipsusr}5: ll handle*
−rw−rw−r−−   1 textdemo   fxalpha      99581 May 26 17:31 handleOUP.log
ds1−pbz{awipsusr}6: more handleOUP.log
. . . .
><><><><><><><><><><><><><><><><><><><><><><><><><><>
<> New Activity Log Started for handleOUP pid:3920 Mon May 26 16:39:44 UTC 2003
><><><><><><><><><><><><><><><><><><><><><><><><><><>
awipsID = KPBZTAFMGW
mapToAfosAndWmoIds():
```

```
        ID = KPBZTAFMGW
        Found match −− awips ID is valid.
        afos ID = CRWTAFMGW
        wmo abbrev ID = FTUS41 KPBZ
product =
/awips/adapt/avnfps/RAP/xmit/pending/014−KPBZTAFMGW−FTUS41−KPBZ−0305261100−AAA−1053965415
getCategory():
Product Category = TAF
getPriority():
Priority = 0
getSiteCommissionStatus():
commission status = 1
createWMOAbbrevHeader():
        wmo abbrev header = FTUS41 KPBZ 261100 AAA
getProductContents():
createAwipsWANHeader():
        AWIPS Id length =
        awips wan header =
FTUS41 KPBZ 261100 AAA   [Return to Top]
TAFMGW
createTargetFile():
        target product pathname = /tmp/014−KPBZTAFMGW−FTUS41−KPBZ−0305261100−AAA−1053965415
storeInTextDB():
        storeCommand =
        textdb −w CRWTAFMGW < /tmp/014−KPBZTAFMGW−FTUS41−KPBZ−0305261100−AAA−1053965415
        Successful storage to text database.
sendWANMsg():
        wanCommand = distributeProduct −d 261100 −p 0 −w AAA −a DEFAULTNCF −c 0  KPBZTAFMGW
/awips/adapt/avnfps/RAP/xmit/pending/014−KPBZTAFMGW−FTUS41−KPBZ−0305261100−AAA−1053965415
        Message to DEFAULTNCF successfully submitted.
isNWWSProduct():
        Product found in NWWS exclude list as CRWTAFMGW.
Product is not an NWWS product.  Not sending product
over NWWS up−link.
archiveOUP():
moveFile():  /tmp/014−KPBZTAFMGW−FTUS41−KPBZ−0305261100−AAA−1053965415 to
/data/fxa/archive/OUP/scratch/014−KPBZTAFMGW−FTUS41−KPBZ−0305261100−AAA−1053965415.20030526_163948
        /tmp/014−KPBZTAFMGW−FTUS41−KPBZ−0305261100−AAA−1053965415 moved successfully...
        Successfully archived product
/data/fxa/archive/OUP/scratch/014−KPBZTAFMGW−FTUS41−KPBZ−0305261100−AAA−1053965415.20030526_163948
Script done....
. . . . .
```

(iii) Check corresponding entry in distributeProduct log file: [Return to Top]

```
ds1−pbz{awipsusr}9: more distributeProduct
. . . . .
16:39:48.463 MHSProduct.C EVENT:
Product request (KPBZTAFMGW) was successfully submitted to the MHS request
server for distribution.
        Message Attributes:
```

Request ID: PBZ−395329
To: DEFAULTNCF
Priority: 0
Type: Routine
Subject:
Handling Action: 0
16:39:48.483 MHSProduct.C EVENT: PBZ−395329
. . . . .

Comments:
The server updates periodically (every 30 s or so) the file xmit/tstamp. The monitor GUI checks the file to determine whether the server is running. If the check time − tstamp update time is less than 1 min, the "Last Pending Check" button is green, If the difference is between 1 and 1 min, the background color changes to orange, if more than 2 min, the button is red. Only one instance of the server should be running at a time. The software checks performs the check before starting avnxmitserv, so multiple clicks on the start button is fine. The tstamp file contains information whether the last transmission succeeded. This is indicated by
the background color of the "Status" button in the Monitor GUI. If no transmission was made, the tstamp file is of length 0, otherwise contains one byte, 'y' if the transmission succeded, 'n' if failed. The background color of the "Staus" button is grey, green or red, respectively. The transmission server invokes handleOUP.pl to transmit a forecast. It does get its return code, however one cannot determine wheter the transmission was succesfull, based on that value. Further investigation is required, as shown above.

NOTE: The tstamp file must be writable by the user. Never restart the server as root.
NOTE: After ds1−ds2 failover swap the transmission server must be restarted. [Return to Top]

Troubleshooting problems with data feed (Missing TAFs, METARs or TWEBs)

(i) Check trigger log files:

ds1−nmtw{awipsusr}12: pwd
/awips/adapt/avnfps/1.0/logs
ds1−pbz{awipsusr}20: ls *trigger*
mtrtrigger_Fri mtrtrigger_Sun mtrtrigger_Wed taftrigger_Sat taftrigger_Tue twbtrigger_Mon twbtrigger_Thu
mtrtrigger_Mon mtrtrigger_Thu taftrigger_Fri taftrigger_Sun taftrigger_Wed twbtrigger_Sat twbtrigger_Tue
mtrtrigger_Sat mtrtrigger_Tue taftrigger_Mon taftrigger_Thu twbtrigger_Fri twbtrigger_Sun twbtrigger_Wed
ds1−pbz{awipsusr}16: cat mtrtrigger_Sun
. . . . .
mtrtrigger 08:45:10: processed /awips/adapt/avnfps/data/mtrs/KDCA/0305250840
mtrtrigger 08:45:35: /data/fxa/trigger/CRWMTREKN No such file or directory
mtrtrigger 08:52:25: processed /awips/adapt/avnfps/data/mtrs/KCLE/0305250851
mtrtrigger 08:52:27: processed /awips/adapt/avnfps/data/mtrs/KPIT/0305250851
. . . . .

The "No such file or directory" entry might have been caused by some local application trigger that moves (instead of copying) files from /data/fxa/trigger. [Return to Top]

Missing AVN MOS data

(i) Check whether AWIPS MOS decoder produces NetCDF files:

```
ds1−nmtw{awipsusr}15: ll /data/fxa/point/mos/AVN/netcdf
total 102170
−rw−rw−rw−  1 fxa      fxalpha   10453024 May 23 04:20 20030523_0000
−rw−rw−rw−  1 fxa      fxalpha   10453024 May 23 16:22 20030523_1200
−rw−rw−rw−  1 fxa      fxalpha   10453024 May 24 04:21 20030524_0000
−rw−rw−rw−  1 fxa      fxalpha   10453024 May 24 16:21 20030524_1200
−rw−rw−rw−  1 fxa      fxalpha   10453024 May 25 04:24 20030525_0000
```

NOTE: Starting with OB2, the data should be available every 6 hours.

(ii) Check whether the decoder is running:

```
as1−nmtw{awipsusr}3: ps −ef | grep AVN
   fxa  2414  2406  0  May 23  ?       5:49 /awips/fxa/bin/DecoderAVN as1−nmtw/4357/2406 1074447384
```

[Return to Top]

Setup or Monitor GUI will not start

(i) On the workstation where the attempt was made to run the GUI, check files

```
ws2−pbz{awipsusr}1: cd /data/logs/fxa
ws2−pbz{awipsusr}2: ll avn*
−rw−rw−r−−  1 textdemo   fxalpha        0 May 16 12:43 avnsetup.log
−rw−rw−r−−  1 awipsusr   fxalpha        0 May 17 11:53 avnwatch.log
```

Length 0 indicates no error messages were written to stdout/stderr. These log files are overwritten each time the GUI is started.

(ii) Start the GUI from command line:

```
lx2−nmtw{awipsusr}1: cd /awips/adapt/avnfps/bin
lx2−nmtw{awipsusr}2: ./avnwatch.sh
lx2−nmtw{awipsusr}3:
```

Any uncaught Python errors will be written to standard error.

It is possible that the Monitor GUI will crash. It may or may not produce a core file, depending on permissions, in the top−level directory /awips/adapt/avnfps/1.0. There were cases reported where crashes were related to METAR data. When new reports arrived, avnwatch would start up succesfully. [Return to Top]

METARs will not display

In the Forecast Editor window, if a forecast is restored from a backup file, the corresponding METARs will not be displayed. There is a workaround:

Before restoring the forecast, first load the same forecast either via the "Amd" or "Cor" buttons in the Monitor GUI, or by invoking "Load" dialog from the Forecast Editor. Then use the "File/Restore" menu to restore the desired version from a backup file. [Return to Top]

AVNFPS Known Bugs

Release 1.0
August 06, 2003

Known Bugs

1. If you bring up the TafGen editor, then hit the restore button that the editor will not setup the listing of Metars on the bottom screen. [Return to Top]
2. Following the issuance of a TAF containing a forecast of significant weather, if the subsequently issued TAF forecasts **no** significant weather, AvnFPS erroneously fails to flag/warn, if the NSW (No Significant Weather) keyword is omitted from a **TEMPO** group. [Return to Top]

AVNFPS Lessons Learned

Release 1.0
June 19, 2003

Lessons Learned

1. The MTR times on the main screen are the times received, not time of the actual observation. This is especially confusing and misleading after an NWSTG outage when old data gets sent as current
2. There is no indication on the main screen of who is logged in. There should be a way to change the forecaster from the main screen, which will load in his/her Xdefaults. Personal Observation: "I can't tell you how many times I've come in the next day to find crazy fonts that other people have changed my account to (thinking they were using their own account)! When that happens, I have to delete /awips/adapt/avnfps/RAP/etc/app−resources/X.4 to revert back to the default."
3. The layout of the AvnFPS buttons are similar, but not the same as that of the Aviation Workstation. AvnFPS will take some minor transition time to understand some of the various new features of the software. It is reasonable to use the AvnFPS for a least a week or two before making a complete evaluation of the system.
4. Upon first starting AvnFPS, each forecaster should set up his/her own AvnFPS monitor and editor Fonts, since the default Font may be too small. Changing the Font style can be accomplished by running the Setup, which is found under the "Options" menu−button.

5. Many offices that used the Aviation Workstation preferred to have the TAFs automatically print upon transmission. The AvnFPS also has this capability and can be toggled to print by running the Setup.

[Return to Top]

AvnFPS versus Aviation Workstation

Release 1.0
June 19, 2003

AvnFPS versus Aviation Workstation

1. The AvnFPS has several additional features not provided in the Aviation Workstation. Some of those new features are listed below:
    1. The AvnFPS allows the forecaster to visually monitor each forecast parameter (wx, vsby, wnd, cig).
    2. The AvnFPS provides a capability to easily monitor the time of the last TAF and METAR.
    3. The AvnFPS provides both visual and audible alerts whenever a new discrepancy between the TAF and METAR occurs.
2. The start up of the AvnFPS monitor screen is slower than the Aviation Workstation, but once the software is up the program run−time is similar.

[Return to Top]

Selecting individual forecasters:

Aviation Workstation you press the down arrow button by the “Forecaster Id:”, then select your forecaster number.

AvnFPS you scroll through list of forecasters and highlight your name, then press the “OK” button.

[Return to Top]

Setting up individual test Font and color preferences:

Aviation Workstation you press the “Preferences” menu−button, then select “Font User Settings”. Select your name from scroll menu, then Font menu will appear. Scroll through desired Font, then select.

AvnFPS you press the “Options” menu−button, then select “Setup”. A scroll listing of setup options will appear, one of these will be Font (used for monitor Font size), Listbox.font, Entry.font and ForecastEditor*Text.font. You press the Font you would like to change, then a scroll of Font size listing will appear. You then select the desired Font size, then press the “Okay” button. A sample text is available on the top of the scroll frame to illustrate the Font size.

Once you are done with selecting Font size, then press the “Save” button, then “Close” button. At this time you will need to restart the program. The restart is accomplished by pressing the “File” menu−button from the main monitor screen, then select “Restart”. You will be prompted to make sure you want to restart.

You will notice that the AvnFPS has greatly expanded the capability to individualize the program.

[Return to Top]

Monitoring TAFs:

Aviation Workstation has two circles that will change color depending on criteria differences between the TAF and Metar. The text box on the right of the circles will alert the forecaster of the weather parameter that is out of tolerance.

AvnFPS color codes each weather parameter for both current and two hours advance depending on criteria differences between the TAF and Metar. By pressing the station identifier button on the left, the program will highlight in red the exact weather parameter that is out of tolerance.

[Return to Top]

Alerting to Old/Missing METAR/TAFs:

Aviation Workstation will color both the current and 2−hour circles grey to represent old/missing Metar/TAF. The text box on the right will provide information which product is not updating.

AvnFPS displays the latest time a TAF and Metar was updated in the program. The background color of the TAF and MTR times will change color to highlight that the product is old/missing.

[Return to Top]

Amend/Correcting TAFs:

The AvnFPS amend and correction buttons are situated exactly like the Aviation Workstation and the overall layout of the editor is similar as well with the TAF editor on top and Metars on the bottom screen. However, there are some differences with the editor.

Aviation Workstation needs a = sign at the end of the TAF.

AvnFPS does not need a = sign at the end of the TAF, but will insert the = sign upon QC’ing or transmission.

[Return to Top]

QC'ing TAFs:

Aviation Workstation will highlight the line of the TAF that is improperly formatted and will a pop−up window with the list of errors and/or warnings. Once there are no errors found, the the pop−up will display Success.

AvnFPS will show improper formatted TAFs by highlighting the text in red or orange. You can obtain a box explaining the reason(s) for the format error by placing your mouse over the highlighted text . Once there are no errors found, the program will show “QC successful” on the very bottom of the editor.

[Return to Top]

Loading TAFs in TAFGen:

You load TAFs in the Aviation Workstation by pressing either the “Prev” button for latest issued TAFs, “Work” button for the saved worked files or “Blank” button to start totally new.

The AvnFPS allows you to load TAFs by pressing the “Load” button, then selecting which products (set of TAFs) you want to edit. Once the TAF sites are selected then press the “OK” button. To edit saved work files, press the “File” menu−button, then select “Restore”. At this time, a file selector will appear. Select the saved file, then press the “Open” button.

A box will appear displaying the contents of the saved file. You can accept the work TAFs by pressing the “Yes” button or cancel by pressing the “No” button.

[Return to Top]

Sending and/or Queuing TAFs in TAFGen:

Aviation Workstation allows you to send your TAFs by pressing the “Xmit” button, then selecting either the “Send Now”, “Send Later” or “Cancel” buttons. Send Now will send the TAFs immediately, while the Send Later option will prompt you with the list of TAFs and time of queue.

AvnFPS allows you to send your TAFs by pressing the “Send” button. A “Time Selection” menu will appear allowing you to set the transmit time. Once you have set the desired time, then press the “OK” button. The background of the TAF editor screen will turn white showing that your TAFs are set to transmit.

[Return to Top]

AvnFPS Tutorial

Release 1.0
June 19, 2003

AvnFPS Tutorial

Karen Oudeman, from WFO JKL, provided all of the information below and includes instructions on the following topics:

To start AvnFPS: [Return to Top]

1. Right click on a text or WS workstation or on the AWIPS icon on Linux…and go to…submenu..." start AVNFPS", then click on "Watch"
2. Give it a minute to start up…it is quickest on Linux and slowest with the text workstations using HPs.
3. Select your name and click on 'OK.' This will help in the verification program and will set your personal fonts, etc.

To Monitor: [Return to Top]

1. Colors are similar to RAVE. To learn more about the colors, go to "file" and click on "view rules".
2. If there is a problem or potential problem, click on the 4 letter ident to view the TAF problem and the matching component in the observation will both be highlighted.

To generate a regularly scheduled TAF: [Return to Top]

1. Click on **TAFGen**

Here are your choices:

A. Write/edit from scratch…delete previous TAF(s) and create your TAF…then move on to step v. below.

Or

B. Click on LOAD
   i. Then usually select "latest" or one of the selections below.
      1. **previous (default)** – loads in previous TAF
      2. **template** – date and time coded for you with the rest of the TAF blank
      3. **merge** – loads in previous TAF along with replacing the first line of the TAF with the current observation, plus any template information that needs to be aded at the end of a TAF
   ii. **Forecast Type** choices:
      1. **Regula**r (default)
      2. **Delayed**
      3. **Correction**
      4. **Amendment**
   iii. Click **OKAY**
   iv. Optional: Click on text to see the latest TAF, MAV output, and METAR observations or click on graph to see the graphical display of all 3.
   v. Edit/create your TAF
   vi. **CHOOSE NAME** – choose your name if yours does not already appear.
   vii. Click on **QC/FORMAT** – this is the equivalent to TAF check, but needs to be done right before you send because it does some formatting required to send.
   viii. Click on **SEND** and when asked "Are you sure?" click 'yes.'
   ix. **PRINT** autoprint is under "options"

To generate an Amendment: [Return to Top]
Here are your choices:

1. Take the steps for generating a regularly scheduled **TAF**, but choose **AMENDMENT** for the **FORECAST TYPE**. This can be used for a single site or multiple sites.

Or

2. Take the short route and click on **AMD** "you'll get a separate window to edit" from the main AvnFPS screen. This will put in your latest TAF with the most current observation. You are now at step **v** above and can create/edit your TAF and continue on with steps **vi** through **ix**.

If you have multiple sites you need to amend, just click **AMD** then hold down the "ctr" button and click on the sites desired. Then click **LOAD.**

To generate a Correction: [Return to Top]
Here are your choices:

1. Take the steps for generating a regularly scheduled TAF, but choose **CORRECTION** for the **FORECAST TYPE**. This can be used for a single site or multiple sites.

Or

2. Take the short route and click on **COR** from the main AvnFPS screen. This will put in your last TAF exactly how you had it. *It will create a Correction for the site you have depressed and is highlighted yellow on the AvnFPS main window.* You are now at step **v** above and can edit your TAF and continue on with steps **vi** through **ix**.

To generate TAFs for Backup Operations: [Return to Top]

1. Click on **TAFGen**
2. Click on **LOAD**
   a. Select/highlight **SERVICE BACKUP** "or whatever that file is called" and, to the right under **SITES**, select/highlight the locations that are being backed up. Then continue as before.

[Return to Top]